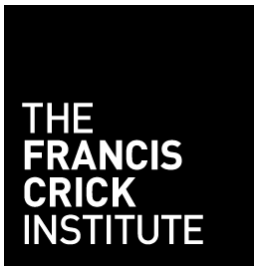
A mathematical analysis of developmental gene regulatory networks

Herrera Delgado, Edgar

Awarding institution:
King's College London



Doctoral Thesis

A mathematical analysis of developmental gene regulatory networks

Edgar Herrera Delgado

King's College London

Mathematics Department

A thesis submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy

Supervisors

Dr James Briscoe, FRS FMedSci

Professor Peter Sollich

November, 2019

Ésta tesis está dedicada a mis padres, Adriana y Edgar.

Acknowledgements

First and foremost I must thank my two supervisors James Briscoe and Peter Sollich. They have guided me in all scientific aspects with their extraordinary expertise, patience and professionalism; always leading by example. Katherine Exelby and Rubén Pérez who have been extremely fun and stimulating to collaborate with, and the Briscoe lab. Thank you to the Francis Crick Institute, King's College London and CONACYT.

There are many people who have been crucial throughout my PhD. To my dear friends in the disordered systems group, Pablo, Ryan, Robin and Davide and many more, for all the Grand Prix and making my time in London so much fun! Thank you to Nerea, it's always great to chat and have "figurative" coffees. Alejandra for your support and insight. Natsumi for helping me in my dream to learn Japanese and for your friendship どうもありがとう。Luciano and Jelena, you always motivate me to remain true to myself. To 湉 Justina for having so many ideas, discussions and adventures together. To my cousin Cuauhtémoc whose enthusiasm and sincere friendship I hold dearly. My grandfather José Ángel (cuyo brío y energía me han motivado a siempre perseverar). To the rest of my family.

Finally, thank you to my parents Adriana and Edgar, for everything.

Abstract

The specification of cell fates is a central aspect of multicellular development. In order to understand this complex phenomenon, mathematical models combined with experimental perturbations have been used. In this thesis we were motivated by the vertebrate neural tube to develop mathematical tools and models that explain the emergence of discrete domains of gene expression as well as the precision of the boundaries between them.

We developed two dimensionality reduction methods based on the Zwanzig-Mori projection that allow a dynamical system to be divided into an arbitrary subnetwork and bulk. The bulk is then replaced by memory functions which correct for the baseline approximation of each method. For the first method the bulk is assumed to be at Steady State; for the second method the bulk is assumed to be at Quasi-Steady State. Using these methods, we were able to accurately capture the dynamics and properties of the original systems. Furthermore, after reducing the dimensionality of the system we show how these methods provide insight into the functioning of biologically relevant networks and we demonstrate how the methods can be used in a complementary way to gain an understanding of the importance of individual reactions in the Gene Regulatory Network (GRN) that patterns the ventral neural tube.

To understand how the boundaries between domains of distinct cell fates are established, we developed a stochastic model of ventral neural tube patterning. This model recapitulated the precision and positioning of wild type and mutant domain boundaries. We modelled perturbations of coding and regulatory regions, and found that the GRN contributes to boundary precision. We performed a computational screen in parameter space that demonstrates that boundary precision can emerge from two dynamical properties that modulate the changes in transition time between steady states in the system. Supported by experimental results, this established design principles for the generation of precise domains between gene expression domains in developing tissues.

Articles contributing to this thesis

1. Herrera-Delgado E, Perez-Carrasco R, Briscoe J, and Sollich P. (2018). Memory functions reveal structural properties of gene regulatory networks. *PLoS Computational Biology*, 14(2):1–25.

2. Herrera-Delgado E, Briscoe J and Sollich P. Memory functions decompose dynamics far from steady state. (In preparation)

3. Exelby K*, Herrera-Delgado E*, Garcia Perez L, Perez-Carrasco R, Sagner A, Metzis V, Sollich P and Briscoe J. (2019). Precision of Tissue Patterning is Controlled by Dynamical Properties of Gene Regulatory Networks. *bioRxiv*, 10.1101/721043.

The work from [1] covers the contents of chapters 2 and 3 and part of introductory chapter 1. The work from [2] covers the majority of chapter 4. The theoretical modelling aspects of [3] with corresponding appendixes cover the contents of chapters 5 and 6.

Contents

Figures

1

Introduction to modelling in development

In this chapter I provide background that will offer context to this thesis. I start with a summary of how individual cells make fate decisions, emphasising the role of gene regulatory networks. I then provide a review of the role of mathematical modelling in aiding biological understanding, focused on developmental biology. I highlight the use of dynamical systems theory as well as stochastic processes in providing novel understanding for biological systems. This is central to this thesis as most of the contributions in this work are underpinned by these approaches. I then introduce the problem of models becoming difficult to understand when they reach greater complexity and the need for

dimensionality reduction techniques. This point is a further important theme of the thesis and I highlight a number of approaches reflecting the state of the art in order to motivate the contents of Chapters 2-4. I give an overview of vertebrate neural tube patterning; this is the system to which I will apply the methods from Chapters 2-4, and which motivates the work in Chapters 5 & 6. Finally, I describe the aims of this thesis.

1.1. Cell fate decisions

1.1.1. Motivation

A central question in understanding multicellularity is how different cell types are established in an organised manner (Gilbert, 2016). Undifferentiated cells divide and differentiate in a coordinated manner to form distinct tissues with specific functions. The organisation of these tissues is established during development through emergent behaviours that shape the developing organism. The mechanisms by which this happens have begun to be understood through studies in embryology and, more recently, advances in molecular biology.

1.1.2. Gene regulatory networks determine cell fates

The fate of a cell is determined primarily by which genes are expressed. This eventual decision depends mostly on transcription factors (TFs), which are proteins capable of binding cis-regulatory regions of DNA to promote or inhibit gene expression. A single TF can bind many different target gene sequences with different affinities, which means each TF can regulate many different genes (Davidson, 2001). Furthermore, multiple TFs can bind to a single regulatory region and act in a cooperative manner, leading to very complex systems (Yu et al., 2003). These systems are defined as Gene Regulatory Networks (GRNs) which can be represented as logic maps that detail the various regula-

tory interactions (Davidson, 2001). At the core of a GRN is a recursive set of regulatory links that forms a transcriptional network, the dynamics of which are responsible for the spatial and temporal patterns of gene expression (Levine and Davidson, 2005).

1.1.3. GRNs function as dynamical systems

In developing organisms, GRNs are capable of producing emergent behaviours that are observable at a tissue scale even though interactions are happening at a molecular level. Famous examples of this are the GAP genes in arthropods, sea urchin endomesoderm specification and vertebrate somitogenesis (Hinman et al., 2003, Jaeger, 2011, Oates et al., 2012). Given the logic representation of GRNs it has been possible to generate mathematical representations, and mathematical modelling has taken centre stage in explaining emergent behaviours in development (Sharpe, 2017).

1.2. Models in developmental biology

In this section I provide some context as to the historical influence of modelling in developmental biology and the importance of GRNs for building precise and principled models.

1.2.1. Historical context

The idea of fundamental principles governing biological systems was famously explored already by Darcy Thompson (Thompson, 1917) and this soon inspired a field of science known as "theoretical biology" (von Bertalanffy, 1928, Waddington, 1940). In the following paragraph I highlight the emergence of two mathematically distinct mechanisms that have proven central to understanding developmental cell fate decisions. These models were proposed before their GRNs and specific molecular elements were properly understood. Despite these unknown factors they have stood the test of time, and as

the details of developmental processes have been uncovered, there is more evidence to support their validity (Briscoe and Kicheva, 2017).

The first mechanism came in 1952, before DNA had been characterised, with Alan Turing's "The Chemical Basis of Morphogenesis", which provided novel intuition as to how asymmetric or repetitive patterns may be generated (Turing, 1952). Reaction-diffusion systems had long been a common theme in the physics literature (Fick, 1855), and had even been applied to biological settings (Kolmogorov, 1937, Tikhomirov, 1991). Turing's work, however, involved a novel mathematical idea regarding instability of steady states induced by diffusion, and provided a unified explanation for a variety of developmental processes.

The other mechanism was conceived by Lewis Wolpert through his proposed solution to what he termed the "French Flag problem" (Wolpert, 1968). He provided the concept of positional information: cells react to a gradient of signal and correspondingly make different decisions depending on the amount of exposure to the signal (Wolpert, 1969). This was proposed as an explanation for cell fate decisions across many developmental tissues, ranging from limb development to lineage specification during gastrulation (Wolpert, 1969).

Importantly, with the aid of technological breakthroughs to study biology, the fundamental principles of the theories by Turing and Wolpert have been proven to be prevalent across a large variety of developmental systems (Green and Sharpe, 2015). The enduring relevance of these two ideas highlights the great insight and influence that rigorous mathematical theory can bring to developmental biology.

1.2.2. Modelling GRNs

The mechanisms proposed by Turing and Wolpert have inspired more complex models that build on the ideas that they put forward. Through the advent of molecular biology

it has become possible to pinpoint which are the key players that constitute different GRNs and how these elements interact with each other. This has led to models that are no longer only phenomenological but where the functional form of the equations, the nodes, and the logic of interactions are based on biological evidence (Bintu et al., 2005, Raser and O'Shea, 2005, Reinitz and Sharp, 1995, Sherman and Cohen, 2012). It is then possible to parametrise these models based on either direct measurements of the parameters (Hammar et al., 2014, Sanchez et al., 2011), or inference from other biological measurements (Cohen et al., 2014, Jaeger et al., 2004).

These GRN based models have been used successfully to explain fundamental questions at a tissue scale or produce precise predictions to be tested experimentally. Examples of this are: patterning scaling (Almuedo-Castillo et al., 2018, Ben-Zvi et al., 2008, Inomata et al., 2013, Umulis et al., 2010), precision of patterning (Kicheva et al., 2014, Lucas et al., 2018) or the relation of oscillations to patterning (Gonze et al., 2005, Lewis, 2003). In each of these cases the mathematical models are fundamental to a correct understanding of the functioning of the respective systems; in most cases they also provide predictive power to motivate experiments.

1.3. Use of mathematical theory for understanding development

In this section I highlight how methods and tools originally developed in mathematical and physical contexts have been successfully used in developmental contexts, providing novel insight as to the functioning of these systems.

1.3.1. Dynamical systems theory

As mathematical models have become more prevalent, dynamical systems theory has been used to elucidate fundamental properties of developmental systems. The concept of stable fixed points has been used to model cell fates based on a cell's gene expres-

sion profile. Multistable systems have then been used to model the possibility of a cell choosing between multiple steady states during development (Murray, 2007). Bifurcation theory has then proven useful to analyse the behaviour of a system; e.g. simply by calculating the steady states of the system and applying Linear Stability Analysis (Naoki et al., 2019, Strogatz, 2014, Wilhelm, 2009). Concrete examples of emergent behaviours that have been characterised with this approach are oscillations, multi-stability and Turing instabilities (Angeli et al., 2004, Gierer and Meinhardt, 1972, Kondo and Miura, 2010, Morohashi et al., 2007). The above methods have also been used to determine whether specific network topologies are capable at all of generating a given behaviour or not (Diego et al., 2018, Leon et al., 2016, Marcon et al., 2016). Other techniques such as analysing vector fields have been used to understand the differences in dynamics between bistable systems and oscillators (Jiménez et al., 2017, Leon et al., 2016), while more detailed analyses of energy landscapes have been crucial to characterise the decision making leading to cell state transitions (Jaeger and Monk, 2014, Verd et al., 2014, Wang et al., 2011).

1.3.2. Stochasticity

As decision making within a cell began to be understood more thoroughly, it soon became evident that intrinsic and extrinsic noise play a fundamental role in this process (Chubb et al., 2006, McAdams and Arkin, 1997, Raser and O'Shea, 2005). These type of stochastic reactions can sometimes be modelled using the "master equation", a first order differential equations that represents probabilistic reactions (Van Kampen, 2007). While this approach provides exactness, it cannot always be applied rigorously and can be inefficient computationally in systems of more than a few molecular species (Van Kampen, 2007). An effective approximation of the master equation is the Langevin equation (developed by Paul Langevin), which has since been adapted to chemical and

biological systems (Anderson and Kurtz, 2011, Gillespie, 2000, 2002, Lemons and Gythiel, 1997). This equation has been shown to be accurate in modelling intrinsic noise in biological settings as long as the number of molecules is not too low; accordingly, it has been used extensively (Grima et al., 2011, Lemons and Gythiel, 1997).

Detailed models of cell fate decisions have thus been developed and they have provided insightful explanations as to the mechanics of how these processes take place (Maamar and Dubnau, 2005, Wang et al., 2009). Other models explore diverse mechanisms in which noise plays a key part, such as noise damping (Koseska et al., 2007) or exploiting noise to make decisions (Kalmar et al., 2009, Maamar et al., 2007, Süel et al., 2006). Overall such models have opened many biologically relevant questions regarding systems that function "in spite of noise" and systems that work "because of noise" (Eldar and Elowitz, 2010, Kærn et al., 2005, Tonn et al., 2019).

1.4. Dimensionality reduction

In this section I highlight the necessity of using dimensionality reduction methods when working with dynamical models in general, but focussing on biological contexts. I then present an overview of methods that allow a reduction to a specific set of variables as this is of specific relevance to this thesis.

1.4.1. Importance of dimensionality reduction

Continuous models based on, for example, ordinary differential equations (ODEs) describe gene regulation in detail and as described above there are well-developed mathematical theories that provide powerful tools to distil the dynamical details of such systems (Mogilner et al., 2006, Murray, 2007). However, a substantial problem for modelling biological systems is that even for relatively small networks, the number of links and the feedback within the system make intuitive understanding difficult to obtain. This is

due to the fact that ODE models often require a large number of parameters, including binding affinities, degradation rates, production rates, etc. (Gunawardena, 2010). While it is often possible to numerically analyse large systems, the power of analytical tools is in many cases complicated or lost due to lack of knowledge of parameters or the complexity of the network (Stuart, 1994). This is true for ODE systems in general, not just in biological settings, and has motivated the generation of tools that reduce the dimensionality and/or complexity of mathematical models to ease their understanding (Rega and Troger, 2005).

1.4.2. Dimensionality reduction approaches

I next highlight a number of techniques that have been developed to reduce the complexity of dynamical models yet preserve specific features of the behaviour of the system. Many of these methods have been used in biological settings though for the most part they are not limited to this context. In the case of mass action reactions, one approach to simplification is to generate intermediate species in a systematic way such as to make different systems comparable. This then allows the identification of motifs that can be simplified. This approach comes with the natural limitation that the species involved cannot be chosen *a priori* and must conform to a particular network structure (Feliu and Wiuf, 2013). Other methods to reduce complexity involve removing nodes or perturbing the system and analysing *a posteriori* the effect that this has on the dynamics of the network. This can be helpful to develop intuition, but generally requires the investigation of a large number of combinations of perturbations (Gay et al., 2010, Okino and Mavrovouniotis, 1998). A further approach, although applicable only in rather specific cases, is to enforce a timescale separation between network nodes, chosen in such a way as not to perturb the dynamics significantly (Sunnaker et al., 2011). One can find "morphisms" that relate network structure to function in order to then simplify a set of

nodes into potentially simpler motifs; this can reveal an interpretation of how certain motifs work but does not provide a direct conversion from the original network (Cardelli, 2014). Less formal methods for replacing well understood network motifs with simpler motifs producing similar behaviour have also been extensively explored (Alon, 2007). Along similar lines, entire parts of a network can be substituted with effective nodes with more complicated dynamics that produce similar output for a set of chosen species, although the feasibility of this depends on network structure and the complexity of the original dynamics (Apri et al., 2012).

1.4.3. Dimensionality reduction approaches on specific variables

Methods of dimensionality reduction where one can arbitrarily choose the variables to be removed provide powerful tools that allow one to test hypotheses more freely. They are of particular importance for this thesis so I highlight some of these approaches in this section. Some examples in this class of methods are: orthogonal decomposition (Berkooz et al., 1993), balanced truncation (Gugercin and Antoulas, 2004) or the Koopman operator (Froyland et al., 2014). An important approach that allows the separation of a dynamical network into subnetwork and bulk observables, where only the subnetwork is then tracked, is the Zwanzig-Mori formalism (Mori, 1965, Nakajima, 1958, Zwanzig, 1961). This method has the advantage of capturing the behaviour of the discarded elements of the system with explicit terms that can be thought of as memory functions see *e.g.* (Ritort and Sollich, 2003). This method can provide an exact reduction, but can become difficult to utilise (Chorin and Stinis, 2006). It was originally developed in the context of Hamiltonian dynamics, and has since been used to study a variety of phenomena including polymer physics (Hijón et al., 2010), dynamical density theory (Wittkowski et al., 2012), biochemical reactions (Rubin et al., 2014) and molecular dynamics (Li et al., 2014). Many extensions of the original approach have been generated,

such as looking at time-convolutionless versions (Tokuyama and Mori, 1976, Venturi and Karniadakis, 2014), or variants for non-Hamiltonian dynamics (Chorin et al., 2000, Zwanzig, 2001). The non-Hamiltonian versions have then been developed further to understand the implications of the unknown initial conditions of the species that are not being tracked (Chorin et al., 2006). These reduced systems are generally hard to characterise explicitly, but solutions have been formulated under some specific conditions (Chorin et al., 2002, Gouasmi et al., 2017).

1.5. Neural tube patterning as a biological example

In this section I provide background as to how the vertebrate ventral neural tube is patterned. This system is a well characterised example of patterning in response to a morphogen gradient (Sagner and Briscoe, 2017), and it will be the central biological case study and motivation used throughout this thesis.

1.5.1. Biological context

The neural tube leads to the eventual ordering and specification of the central nervous system. In order to specify the diverse cell fates in the nervous system, it is patterned along the anterio-posterior and dorso-ventral axes (Dessaud et al., 2008, Papalopulu and Kintner, 1996, Sasai et al., 2014). The molecules and mechanisms involved in patterning the dorso-ventral axis have been extensively characterised. This axis is patterned by discrete domains of TF expression which lead to the neural progenitors that will eventually make up the central nervous system (Alaynick et al., 2011). On the dorsal side of the neural tube, a tissue known as the roof plate emerges from previous developmental processes; the roof plate secretes the morphogen BMP. BMP diffuses ventrally, generating a gradient of activity which results in discrete gene expression domains (Chizhikov and Millen, 2004).

On the ventral side of the neural tube, a tissue known as the notochord, which is also established through previous developmental processes, secretes the morphogen Sonic Hedgehog (Shh). Shh diffuses dorsally into the ventral-most part of the neural tube, inducing it to become what is known as floor plate; the notochord and the floor plate both secrete Shh which in turn diffuses dorsally into ventral regions of the neural tube. This process creates a gradient of Shh which signals into the cells by binding the Shh receptor Patched (Ptch1). The binding of Ptch1 by Shh leads to the eventual activation of the downstream Gli transcriptional effectors (Sagner and Briscoe, 2017). In the absence of extracellular Shh, Gli functions as a repressor; when Shh is present, the downstream signalling pathway modifies Gli proteins turning them into activators (Cohen et al., 2015). The dynamics of Gli are not straightforward as there are three Gli proteins: Gli1 and Gli2 act predominantly as activators, whereas Gli3 acts as a repressor (Cohen et al., 2015). The balance of active and inactive Glis present in a cell lead to a net Gli activity which serves as an input to a GRN. This GRN interprets the net Gli activity and as a result forms the TF expression domains along the dorso-ventral axis of the ventral neural tube (Fig. 1.1A & Fig. 3.2B) & (Dessaud et al., 2010).

1.5.2. Neural tube GRN

The GRN that interprets the graded levels of Gli activity in the ventral neural tube has been characterised through over-expression, knock downs and knock outs of the relevant TFs (Briscoe et al., 2000, 1999, Dessaud et al., 2008, Ericson et al., 1997). The molecular mechanisms that drive regulation between TFs have been elucidated through the use of CHIP-seq and ATAC-seq (Kutejova et al., 2016, Metzis et al., 2018, Oosterveen et al., 2012, Peterson et al., 2012). The GRN is made up of four TFs: Pax6, Olig2, Nkx2.2 and Irx3. It consists of transcriptional repressions that ensure that the cells do not co-express TFs that are unique to a domain (Fig. 1.1B & Fig. 3.2C).

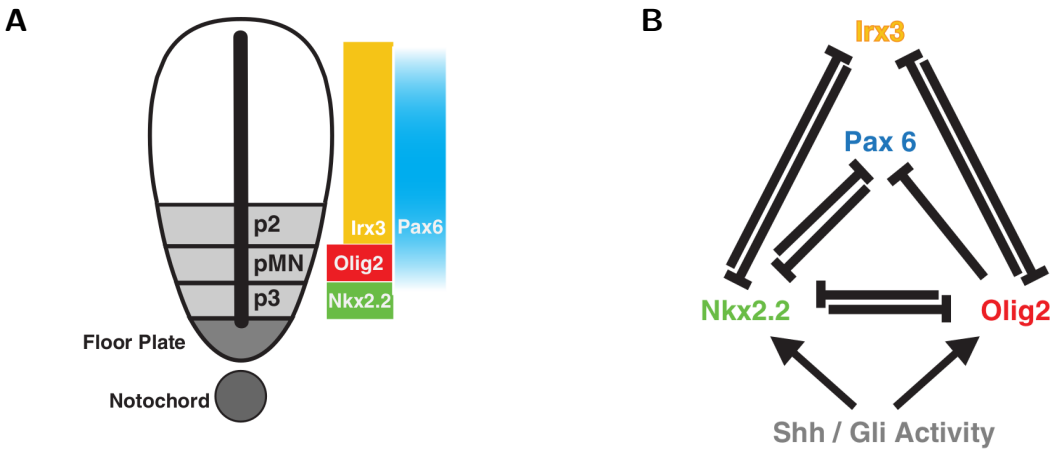


Figure 1.1: Images and captions of this figure are adapted from (Cohen et al., 2014). (A) In the neural tube, the notochord and floor plate provide a source of Shh. This gradient acts through Gli proteins to produce a spatial pattern of distinct progenitor domains in the ventral region (the three ventralmost domains are shown: p2, pMN, p3). Each domain is characterised by the expression of different sets of TFs; the spatial distribution of four of these (Nkx2.2, Olig2, Irx3 and Pax6) is illustrated. (B) Nkx2.2, Olig2, Irx3 and Pax6 form a cross-repressive network with the regulatory links indicated in the diagram. Pax6 and Irx3 are expressed in the absence of any Shh signal. Nkx2.2 and Olig2 are regulated by Gli activity.

The domains are determined as follows: p3 expresses only high levels of Nkx2.2, pMN expresses high levels of Olig2 and medium levels of Pax6, p2 expresses high levels of both Irx3 and Pax6 (Fig. 1.1A & Fig. 3.2B). An interesting behaviour observed at a tissue scale are a series of sequential ventral to dorsal waves of gene expression (Jeong and McMahon, 2005). These waves halt at a fixed position which is determined by the concentration of morphogen, thus establishing the boundaries between progenitor domains. As a consequence of ventral cells transiently expressing genes corresponding to more dorsal domains, cell identities that correspond to higher morphogen concentrations take a longer time to be established (Briscoe and Small, 2015). In mice this process takes approximately 48 hours, afterwards cells differentiate into neurons and migrate laterally out of the tissue (Briscoe and Small, 2015, Jeong and McMahon, 2005).

1.5.3. Neural tube modelling

As a result of the thorough characterisation of neural tube patterning described above, many advances have been made to model and understand the patterning of the neural tube. Equally important, this has lead to a greater understanding of patterning mechanisms which are likely to be broadly present throughout many developmental systems. Through the application of classic dynamical systems theory, an explanation of observed hysteresis in this system was provided (Balaskas et al., 2012). A very thoroughly characterised model was developed by using Approximate Bayesian Computation inference, wild type (WT) and mutant phenotypes being used as targets for the corresponding classifier. The resulting model has tightly constrained parameters which can replicate the positioning of boundaries between domains of the WT and mutant systems (Cohen et al., 2014). This model is additionally able to reproduce the sequential wave of gene expression described in the above section despite not being specifically parametrised to do so. These results suggest that such behaviour is encoded into the network and that it emerges from the model because the system is correctly parametrised. This sequential wave of expression has been further explored to determine if this type of behaviour can be found generally as a property of a patterning system (Rothschild et al., 2016). A further purely theoretical result of great relevance that was motivated by the observed wave of gene expression is the effect of intrinsic noise on boundary position placement and its timing (Perez-Carrasco et al., 2016). Additional work has been done to understand the signalling dynamics of Shh which is likely to have an effect on the patterning of the tissue (Cohen et al., 2015, Junker et al., 2014, Li et al., 2018).

1.5.4. AC-DC

An interesting conceptual finding when modelling the basic structure of the ventral neural tube GRN was that this network can easily transition between oscillatory and multistable

behaviour (Panovska-Griffiths et al., 2013). This network topology was coined the AC-DC motif as an analogy to the behaviours of alternating and direct electric currents. Subsequent models of the neural tube patterning networks have included the AC-DC motif and it has been carefully parametrised in these subsequent studies (Balaskas et al., 2012, Cohen et al., 2014). This motif provides a simple switch between dynamical behaviours, and has also been shown to be able to produce excitable behaviour under noisy conditions (Perez-Carrasco et al., 2018). The AC-DC motif has been found to be relevant for understanding the GAP gene network, as it acts in a modular way to specify domains in response to upstream morphogen gradients (Verd et al., 2019). This finding is intuitive as the GRN responsible for arthropod presents many similarities to the neural tube network, namely mutual cross-repressions and a response to a graded signal. The AC-DC motif has been proposed as part of an explanation to a long-standing question as to how there are two forms of patterning with similar structure but dramatically different dynamics in arthropod species (Verd et al., 2019). Those two dynamics are: sequential as in short-germ insects and simultaneous as in long-germ insects (Clark, 2017, El-Sherif et al., 2012). The AC-DC motif provides a unifying explanation for both behaviours (Verd et al., 2019). This is due to the ease with which it can switch from oscillatory to multistable dynamics with relatively small changes in its parameters while keeping the same topology (Verd et al., 2019). These studies demonstrate the versatility of the AC-DC motif which arose from mathematical modelling but has provided a mechanistic answer to biological behaviours. The variety of behaviours produced by this motif also emphasise the importance of going beyond cataloguing network topologies, but to also study a system's dynamical properties (Jaeger and Monk, 2015).

1.6. Aims

The present thesis is structured as follows. In Chapter 2 I generate a projection approach using the Zwanzig-Mori formalism where one projects memory effects onto a specific set of arbitrary variables. This result necessitates an expansion around a steady state and therefore has limited accuracy when far from the corresponding steady state. The resulting memory functions are however relatively simple and thus provide an intuitive understanding of the effect of memory near steady state. In Chapter 3 I apply this method to a bistable switch and to the well characterised neural tube network. Through interpreting the resulting memory functions I go on to find the relative unimportance of a particular interaction for generating the correct steady states yet its importance for robustness to changes in concentration.

In Chapter 4 I develop a nonlinear projection approach based on the work from (Chorin et al., 2000), which as opposed to Chapter 2 is based on a QSS assumption for the bulk that the resulting memory functions then correct for. This approach is extremely precise even for strongly nonlinear systems and can track oscillations as well as recapitulate multistable systems. The solution of the nonlinear memory function is in itself a substantial contribution and is thus detailed extensively within the chapter. I demonstrate the method's accuracy on a variety of systems as well as on the neural tube network while highlighting its complementarity to the method derived in Chapter 2.

In Chapter 5 I go on to focus on the stochastic aspect of biological patterning. I build on previous work to develop a stochastic system of equations to understand the emergence of boundary precision in the neural tube. Drawing on experimental work performed by collaborators I generate a theoretical explanation for the emergence of boundary precision as a consequence of a GRN. In Chapter 6 I explore generally how a GRN may generate sharpness in a network interpreting a morphogen gradient and find

two mechanisms that could potentially be present across many biological systems. A topological analysis provides networks that most favour boundary precision and include the network used in Chapter 5.

In Chapter 7 I summarise my results, discuss their importance and contribution to different scientific fields and elaborate upon possible future projects that could be pursued on the basis of my results.

2

Memory around a fixed point

2.1. Introduction

We focus in this chapter on reducing model complexity by tracking a subnetwork that is embedded within the remainder – or "bulk" – of a larger system. Analysing the behaviour of the subnetwork and its interaction with the bulk can help reveal and rationalise the properties of the entire network. With any coarse graining approach, including those discussed in Sec. 1.4, a compromise must be reached between the precision of the method, *i.e.* how well it captures the dynamics of the full network, and the simplicity and interpretability of the resulting description. For example, the introduction of

intermediate species can produce model reductions with high precision but these may not be interpretable (Feliu and Wiuf, 2013), while morphism approaches provide interpretable insights into the structure but capture the full model dynamics only for specific initial conditions and system parameters (Cardelli, 2014). We consider important for interpretability the ability to *choose* a subnetwork, guided by biological relevance or the availability of data; none of the methods described above allow this and instead identify a subnetwork based on their internal criteria.

One class of methods that offers the potential to balance accuracy and interpretability are Zwanzig-Mori projections (Mori, 1965, Zwanzig, 1961). Originally developed to allow the extraction of macroscopic equations from a microscopic description of the dynamics – a brief overview can be found in *e.g.* (Ritort and Sollich, 2003) – these methods have since been used to separate a network into an arbitarily chosen subnetwork and bulk in ways that preserve substantial features of the original temporal dynamics (Rubin et al., 2014, Rubin and Sollich, 2016). Used in this way, the approach describes the concentration of components in the subnetwork in detail, while the activities of the species in the bulk are replaced with so-called 'memory functions'. These memory functions are derived from the detailed kinetic description of the remainder of the network and summarise how, by acting via the bulk, the past behaviour of the subnetwork influences its current state (Fig. 2.1). The resulting memory functions are functions of time difference, describing the amplitude of a signal returning from the bulk some specified time after the original signal left the subnetwork. Additionally, it is possible to separate each memory function in order to analyse through which bulk species the signal is flowing, thus providing a means to gain a more systematic understanding of a system.

Evolving from its original applications in critical dynamics and supercooled liquids near the glass transition (Gotze and Sjogren, 1992), this approach has been applied to biochemical networks (Rubin et al., 2014), where it has been used to analyse the dynamics

of signaling in the EGFR (epidermal growth factor receptor) network. It has since been developed further to include also enzymatic Michaelis-Menten reactions (Rubin and Sollich, 2016). Here we apply the Zwanzig-Mori approach to the analysis of GRNs. We derive a general method that is applicable to any thermodynamic state ensemble representation of protein dynamics including transcriptional networks. These thermodynamic equations mechanistically represent interactions between proteins via transcriptional regulation (Bintu et al., 2005) and can be expanded into mass action reactions to which the Zwanzig-Mori projection formalism can be applied. This approach not only justifies a heuristic method where the thermodynamic equations are expanded to quadratic order around a steady state to obtain a set of effective unary and binary reactions, but also opens up the possibility of developing more advanced projection approximations. We test the method on a cross repressive motif, where we show that we are able to qualitatively reproduce the dynamics of the system and obtain a simple intuitive interpretation of the memory functions. Finally, we use the approach on a GRN from the vertebrate neural tube, a well characterised developmental GRN composed of four interacting transcription factors (Cohen et al., 2014). This reveals the importance of specific links within the network and identifies features that appear to be present primarily to increase the robustness of the network to initial conditions, rather than maintain the steady states. Taken together, the study provides evidence that Zwanzig-Mori projections are a powerful and efficient method to simplify and explore the behaviour of gene regulatory networks, with memory functions providing new tools for probing the dynamics near steady state.

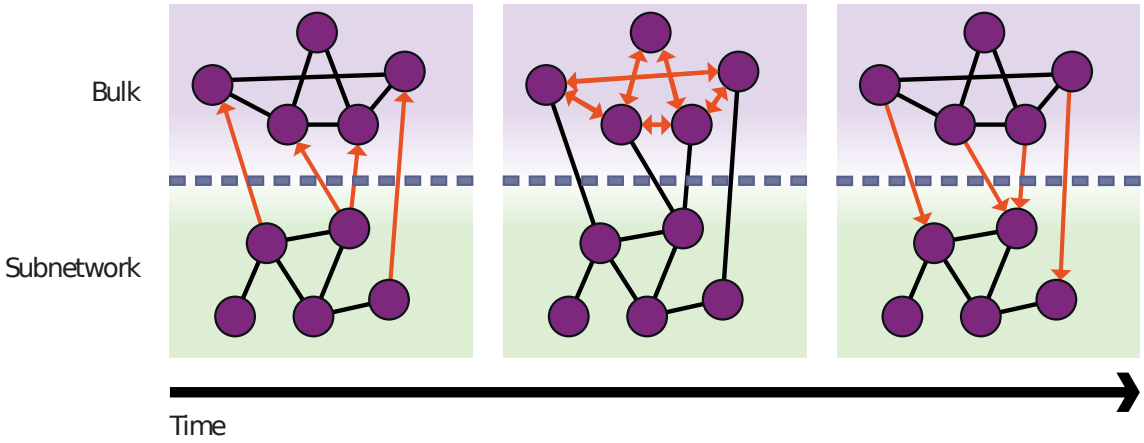


Figure 2.1: Sketch describing the concept of memory within a reaction network. A network is divided into bulk and subnetwork, x axis represents time. Concentration changes in the subnetwork act as signals that leave the subnetwork and travel into the bulk. There they interact with other bulk species and return at a later timepoint via bulk-to-subnetwork interactions. The net effect of such interactions is thus that the subnetwork reacts to its own past. The precise influence of past subnetwork states is governed by *memory functions* that depend on the time difference, *i.e.* on how long ago the relevant signal has left the subnetwork.

2.2. Thermodynamic state ensemble models for gene regulation dynamics

Several classes of model have been developed to describe gene regulation. One such class consists of thermodynamic state ensemble models in which equations are constructed that relate interactions between key components of the regulatory mechanism to gene expression (Bintu et al., 2005, Shea and Ackers, 1985, Sherman and Cohen, 2012). In these models, all possible states of a regulated gene are enumerated. Each state consists of a specific set of transcription factors bound to a DNA cis-regulatory element, weighted by the affinity of interactions. The rate at which a protein is produced is then represented as a ratio of the weight of the subset of states that promote gene expression, to the total weight of all states. We define the vector $\boldsymbol{x}$ such that it contains the protein concentrations for each gene. We can then generically define the time evolution for any

protein concentration, indexed by j, as

$$\partial_t x_j = \sum_{\boldsymbol{n}} x_{(j,\boldsymbol{n})} \alpha_{(j,\boldsymbol{n})} - x_j \beta_j \tag{2.1}$$

$$x_{(j,\boldsymbol{n})} = \frac{w_{(j,n)} \prod_i x_i^{n_i}}{\sum_{n'} w_{(j,n')} \prod_i x_i^{n'_i}} \tag{2.2}$$

Here β_j is the decay rate of protein j. We use the summation over $\boldsymbol{n}$ to represent all possible binding conformations for the DNA that produces protein j. The term $\alpha_{(j,\boldsymbol{n})}$ represents the production rate of protein j from a particular DNA conformation $\boldsymbol{n}$. The factor $w_{(j,\boldsymbol{n})}$ is the specific affinity for individual proteins to the DNA that produces species x_j. Finally $x_{(j,\boldsymbol{n})}$ is the concentration of DNA producing protein j and in conformation $\boldsymbol{n}$. This is assigned in (2.2) as the weight of the conformation, dependent on the protein concentrations in a mass-action form, normalised by the total weight of all conformations. This ensures that $x_{(j,\boldsymbol{n})}$ lies in the range $[0, 1]$, so it is not an absolute concentration; any overall DNA concentration scale is therefore to be understood as incorporated into the protein production rates $\alpha_{(j,\boldsymbol{n})}$. Accordingly one can also think of $x_{(j,\boldsymbol{n})}$ as the probability of finding a certain conformation $\boldsymbol{n}$ of the DNA producing protein j. As made explicit in (2.2), the DNA conformation label $\boldsymbol{n}$ is a collection of integers n_i counting how many copies of protein i are bound to the DNA. This is limited by the number of binding sites for each protein and the compatibility of different proteins to be bound at the same time.

2.3. Baseline of projection method

The Zwanzig-Mori method in general starts from the choice of a set of observables for which dynamical equations are to be obtained. Initially we select, as the simplest possible observables, the deviations from steady state of the concentrations of the chosen subnetwork species. For these observables the Zwanzig-Mori projection leads to a set of

dynamical equations of the form

$$\partial_t \delta x_i = \sum_{j=1}^{N^s} \delta x_j \Omega_{ji} + \int_0^t dt' \left(\sum_{j=1}^{N^s} \delta x_j(t') \underbrace{M_{ji}(t-t')}_{\text{Memory function}} \right) + r_i(t) \tag{2.3}$$

Here δx_i are the subnetwork concentration deviations as defined explicitly below and N^{s} indicates the number of subnetwork species. The Ω_{ji} define a *rate matrix* $\mathbf{\Omega}$ that represents subnetwork interactions. The $r_i(t)$ are so-called *random forces* that arise from the uncertainty about the initial (time $t = 0$) state of the bulk concentrations. (For the linear dynamics they can be expressed in closed form (Rubin et al., 2014) but this is not useful in our context as discussed in Sec. 2.5 below.)

The key quantities in the equations (2.3) are the *memory functions* $M_{ji}(\Delta t)$, which describe how past concentration fluctuations $\delta x_j(t')$ influence the current time evolution. However, while the projected equations are formally exact, it is in general impossible to evaluate these memory functions explicitly. One scenario where this *can* be done is a network of unary reactions, where the full reaction equations are linear in the concentrations (Rubin et al., 2014). This suggests a heuristic approach to calculating memory functions in the current context, which is to linearize the thermodynamic equations. One can think of this as generating a set of effective unary reactions (at least loosely; see Sec. 2.6). Explicitly, one linearly expands the time evolution equations (2.1) with (2.2) inserted. The expansion is performed around a steady state with concentrations y_i, *i.e.* in terms of the deviations $\delta x_i = x_i - y_i$. The r.h.s. of (2.1) then contains only linear terms in the δx_i since constant terms cancel out because of the steady state condition. One can therefore write the time evolution equations in matrix form (Rubin et al., 2014)

$$\partial_t \delta x_i = \sum_{j=1}^{N^{\mathrm{s}}} \delta x_j L_{ji} \tag{2.4}$$

for an appropriately defined matrix $\boldsymbol{L}$ with entries L_{ji}.

2.4. Projected equations for linearised dynamics

From the matrix $\boldsymbol{L}$ one can calculate the terms in the (linear) projected dynamics of the subnetwork concentrations from (2.3) as shown in (Rubin et al., 2014). We summarize the method here. One separates the set of all (linear in concentration) observables into subnetwork (s) and bulk (b), and assumes as above that these concentration deviations δx_i are numbered so that the first N^{s} are the subnetwork observables while the rest are the bulk observables. The matrix $\boldsymbol{L}$ for the linearised dynamics then separates into blocks according to the different observables:

$$\boldsymbol{L} = \left(\begin{array}{c|c} \boldsymbol{L}^{\mathsf{S},\mathsf{S}} & \boldsymbol{L}^{\mathsf{S},\mathsf{B}} \\ \hline \boldsymbol{L}^{\mathsf{B},\mathsf{S}} & \boldsymbol{L}^{\mathsf{B},\mathsf{B}} \end{array}\right) \tag{2.5}$$

where labels S and B represent respectively the set of species in the subnetwork and in the bulk. We use the labels S and B here to represent subnetwork and bulk species respectively, this will be useful once more subnetwork and bulk observables arise (see Sec. 2.7). The rate matrix $\boldsymbol{\Omega}$ with entries Ω_{ji} in the projected equations (2.3) is then simply the subnetwork block of $\boldsymbol{L}$,

$$\boldsymbol{\Omega} = \boldsymbol{L}^{\mathsf{S},\mathsf{S}} \tag{2.6}$$

The memory functions similarly form the elements of a memory matrix, which can be expressed in terms of the blocks of $\boldsymbol{L}$ as (Rubin et al., 2014)

$$\boldsymbol{M}(\Delta t) = \boldsymbol{L}^{\mathsf{S},\mathsf{B}} e^{\boldsymbol{L}^{\mathsf{B},\mathsf{B}}\Delta t} \boldsymbol{L}^{\mathsf{B},\mathsf{S}} \tag{2.7}$$

This form allows straightforward evaluation of the memory functions starting from any given matrix $\boldsymbol{L}$ describing the linearised dynamics. The intuition behind (2.7) is that $\boldsymbol{L}^{\mathsf{S},\mathsf{B}}$

captures signals leaving the subnetwork into the bulk, $e^{\boldsymbol{L}^{\mathrm{B,B}}\Delta t}$ captures the propagation of such signals over time in the bulk, and $\boldsymbol{L}^{\mathrm{B,S}}$ captures the return of the signals into the subnetwork.

2.5. Nonlinear projected equations

To capture the leading nonlinear corrections to the subnetwork dynamics, it is natural to enlarge the set of subnetwork observables by adding quadratic observables, i.e. products of concentration deviations. The projected equations then become

$$\begin{aligned}\partial_t \delta x_i = &\sum_{j=1}^{N^s} \delta x_j \Omega_{ji} + \sum_{1\le j\le k\le N^s} \delta x_j \delta x_k \Omega_{(jk)i} \\ &+ \int_0^t dt' \left(\sum_{j=1}^{N^s} \delta x_j(t') \underbrace{M_{ji}(t-t')}_{\text{Linear memory}} + \sum_{1\le j\le k\le N^s} \delta x_j(t')\delta x_k(t') \underbrace{M_{(jk)i}(t-t')}_{\text{Nonlinear memory}} \right) \\ &+ r_i(t)\end{aligned} \tag{2.8}$$

This form is analogous to (2.3) but now includes the quadratic observables that we have retained, both in the subnetwork interactions (rate matrix terms) and in the memory terms. Once again $r_i(t)$ represents the *random forces* but in this nonlinear case further characterization would require knowledge of the statistics of the initial bulk fluctuations, which is not generally available. We therefore disregard these terms, making (2.3) a *closed* system of equations for the time evolution of the subnetwork concentrations. Note that neglecting the random forces is equivalent to assuming that the bulk concentrations are at their steady state values at the initial time. Formally these initial bulk concentrations should be random rather than deterministic to make the Zwanzig-Mori projection well defined; we follow the strategy of (Rubin et al., 2014) here and use initial Poisson distributions in the limit of vanishing variance. This limit has already been taken in the results above (Rubin et al., 2014). How large protein copy numbers have to be

to allow stochastic effects to be safely neglected is something that could be studied in future work; the answer will depend, among other things, on the degree of nonlinearity of the GRN time evolution equations (Thomas et al., 2012b).

Moving on to the nonlinear memory functions, we are again faced with the difficulty that these cannot in general be evaluated explicitly. However, previous work (Rubin et al., 2014) has developed a systematic approximation technique, which is applicable to the case of reaction networks with unary and binary reactions described by mass action kinetics. In such networks the time evolution equations for linear observables have linear and quadratic terms on the r.h.s. Inserting these into the equations for *quadratic* observables using the product rule gives *e.g.*

$$\partial_t(\delta x_1 \delta x_2) = (\partial_t x_1)\delta x_2 + \delta x_1(\partial_t \delta x_2) \tag{2.9}$$

and one sees that cubic terms arise. The approximation in (Rubin et al., 2014) neglects these terms in the spirit of an expansion to second order in the changes from steady state δx. These changes are therefore implicitly assumed to be small. Defining a vector $\boldsymbol{z}$ that collects all δx variables and their products such as δx_1^2, $\delta x_1 \delta x_2$, one can now again write a matrix form of the time evolution equations:

$$\partial_t z_\alpha = \sum_\beta z_\beta L_{\beta\alpha} \tag{2.10}$$

The rate matrix entries and memory functions can then be calculated (Rubin et al., 2014) from the general formulae (2.6,2.7), applied to the expanded $\boldsymbol{L}$ matrix constructed as explained above. Because the subnetwork block S now contains linear $\{\mathrm{s}\}$ and quadratic observables $\{\mathrm{ss}\}$, the matrices have the corresponding block structures. The coefficients Ω_{ji} are collected in the block $\boldsymbol{\Omega}^{\mathrm{s,s}}$, for example, and the $\Omega_{(jk)i}$ in the block $\boldsymbol{\Omega}^{\mathrm{ss,s}}$ of the overall rate matrix $\boldsymbol{\Omega}$. The memory functions $M_{ji}(\Delta t)$ and $M_{(jk)i}(\Delta t)$ are similarly

contained in blocks $\boldsymbol{M}^{\mathrm{s,s}}$ and $\boldsymbol{M}^{\mathrm{ss,s}}$ of the memory matrix $\boldsymbol{M}$.

To apply the above technique in order to derive nonlinear projected subnetwork equations for GRN dynamics, the obvious heuristic route is again to expand the dynamical equations (2.1), this time to second order in the δx_i, to obtain an effective set of unary and binary reactions. This set defines the matrix $\boldsymbol{L}$, from which the rate matrix and memory functions can then be found as explained above.

2.6. Justifying the heuristics: network expansion

The heuristic method set out above for deriving linear or nonlinear subnetwork equations for GRN dynamics is a useful practical recipe, but its mathematical basis is not obvious. One difficulty is that the thermodynamic GRN equations (2.1) involve rational functions of the concentrations such as $1/(1+x_1)$, and while these can be Taylor expanded they have only finite radius of convergence. Not only does the heuristic method have to throw away an infinite number of higher order terms when truncating the Taylor expansion after linear or quadratic order, but it may implicitly be using the resulting approximation outside the regime where the expansion is even convergent. In addition, while we have described the expanded time evolution equations as representing networks of effective reactions, such an interpretation is in no way assured as there is no underlying mechanistic model and hence no constraints that would ensure an appropriate stoichiometry or even positivity of rate constants.

Our mathematical contribution, in this chapter, is to demonstrate that the above limitations can be resolved, and the heuristic approach justified, by initially *expanding* the thermodynamic equations (2.1 & 2.2) into a network of well-defined unary and binary reactions. The way to achieve this is already suggested by (2.2): we represent each possible existing DNA conformation as an additional species whose concentration has its own dynamical evolution, such that in an appropriate limit of fast DNA binding and

unbinding, it reaches the quasi-steady state (QSS) values of the DNA conformation concentrations (2.2). The prefix "quasi" refers to the fact that this steady state is for given protein concentrations, which themselves change slowly in time. We can then apply the Zwanzig-Mori method to this network to extract memory functions for the chosen subnetwork. In doing this we can justify removing higher order terms as in (Rubin et al., 2014) as the system is composed of only first and second order reactions. Finally the limit of fast DNA binding and unbinding has to be taken. Our result is that this procedure results in exactly the same projected equations as the heuristic approach described above, thus putting the method on a firm mathematical footing. The conclusion applies not only in the context of GRNs but in all networks with the appropriate timescale separation, provided that all fast observables are treated as part of the bulk. More sophisticated projection methods can then be derived by retaining some fast observables (in our case, subnetwork DNA) in the subnetwork. For the case of Michaelis-Menten dynamics, previous work has shown that such an approach allows one to derive projected equations that retain all subnetwork nonlinearities (Rubin and Sollich, 2016). More generally, by embedding GRN models based on thermodynamic state ensembles into the broad class of mass action reaction networks, our network expansion method opens up the possibility of incorporating stochastic effects and of applying a broad range of other approximation and model reduction techniques (Schnoerr et al., 2017).

2.7. Mathematical results

In this section we use a network expansion technique to provide a justification for the heuristic approach to finding memory functions for GRNs that was described in Secs. 2.4 and 2.5 above.

2.7.1. Network expansion applied to a protein-DNA binding mechanism

Our network expansion approach necessitates defining new reaction rates for binding and unbinding of protein to/from DNA that are consistent with the affinities w from the thermodynamic description. To be explicit, we define $k^{p+}_{(j,\boldsymbol{n})}$ as the rate constant for protein p to bind to DNA coding for protein j and in binding conformation $\boldsymbol{n}$, with a similar meaning of unbinding for $k^{p-}_{(j,\boldsymbol{n})}$. These rates will thus describe the interaction each DNA species has with the protein species that bind to it. With these rate constants we write down an equation for the time evolution of each DNA concentration $x_{(j,\boldsymbol{n})}$:

$$\partial_t x_{(j,\boldsymbol{n})} = \sum_p \left(k^{p+}_{(j,\boldsymbol{n}-\boldsymbol{e}_p)} x_{(j,\boldsymbol{n}-\boldsymbol{e}_p)} x_p - k^{p+}_{(j,\boldsymbol{n})} x_{(j,\boldsymbol{n})} x_p + k^{p-}_{(j,\boldsymbol{n}+\boldsymbol{e}_p)} x_{(j,\boldsymbol{n}+\boldsymbol{e}_p)} - k^{p-}_{(j,\boldsymbol{n})} x_{(j,\boldsymbol{n})} \right) \tag{2.11}$$

The first term on the r.h.s. of (2.11) represents binding of protein p to DNA conformation $(j, \boldsymbol{n}-\boldsymbol{e}_p)$ to make DNA conformation $(j, \boldsymbol{n})$. Here $\boldsymbol{e}_p$ is a unit vector whose i-th entry is δ_{ip}, *i.e.* a 1 at position p and zeros elsewhere. Similarly the third term on the right of (2.11) describes unbinding of protein p. The other two terms capture how DNA in conformation $(j, \boldsymbol{n})$ can be lost by these two processes.

We next establish the relationship between the rate constants k in the mass action representation (2.11) and the affinities w of the thermodynamic form. The rate constants need to be chosen to ensure the QSS (2.2) of the DNA concentrations. The simplest assumption is detailed balance, which requires that in this steady state the net change in DNA concentration from any *individual pair* of binding and unbinding reactions between two conformations vanishes. Detailed balance means, in particular, that where there are multiple reaction paths from one DNA conformation to another, which might be distinguished by the order in which the various proteins bind, each path carries zero reaction flux rather than having a preferred forward or backward reaction direction.

Balancing in this way the net change from binding of protein p to DNA in conformation $(j, \boldsymbol{n})$ with the reverse unbinding transition, for the QSS concentrations (2.2), gives

$$k^{p+}_{(j,\boldsymbol{n})} x_p w_{(j,\boldsymbol{n})} \prod_i x_i^{n_i} = k^{p-}_{(j,\boldsymbol{n}+\boldsymbol{e}_p)} w_{(j,\boldsymbol{n}+\boldsymbol{e}_p)} \prod_i x_i^{n_i+\delta_{ip}} \tag{2.12}$$

The protein concentrations cancel from this as they should and we are left with a constraint on the rate constants:

$$\frac{k^{p+}_{(j,\boldsymbol{n})}}{k^{p-}_{(j,\boldsymbol{n}+\boldsymbol{e}_p)}} = \frac{w_{(j,\boldsymbol{n}+\boldsymbol{e}_p)}}{w_{(j,\boldsymbol{n})}} \tag{2.13}$$

To finish our construction, we need to ensure that the DNA concentrations reach the QSS values that we have imposed above with the detailed balance constraint (2.13). This requires that we make all DNA binding and unbinding reactions fast. Formally we introduce a fast rate factor γ and write all rate constants in the form $k^{p\pm}_{(j,\boldsymbol{n})} = \gamma \tilde{k}^{p\pm}_{(j,\boldsymbol{n})}$, with the understanding that we will take the limit $\gamma \to \infty$ at fixed $\tilde{k}$. This construction is analogous to the one previously used for Michaelis-Menten reactions (Rubin and Sollich, 2016). As there, we also need to ensure that the changes to protein concentrations from binding to DNA are negligible, as such effects are not described by the original thermodynamic equations (2.1). (We do, however, include them in the finite-γ numerics of Fig. 3.1B.) This is achieved by scaling all DNA concentrations as $x_{(j,\boldsymbol{n})} = \tilde{x}_{(j,\boldsymbol{n})}/\gamma'$. The intuition for this construction is that the amount of protein bound to DNA is, at most, of the order of the total concentration of DNA, which vanishes for $\gamma' \to \infty$. Finally, to compensate for the DNA concentration scaling, protein production rates have to be scaled as $\alpha_{(j,\boldsymbol{n})} = \gamma' \tilde{\alpha}_{(j,\boldsymbol{n})}$.

With the above rescalings, the limit $\gamma \to \infty$ ensures that the "fast" species, *i.e.* the DNA concentrations are always in QSS with respect to the "slow" protein concentrations. The limit $\gamma' \to \infty$, on the other hand, means that the effects of binding to, and unbinding

from, DNA can be neglected in the equations for the protein concentrations. These statements can be shown mathematically along the lines of the arguments in (Rubin and Sollich, 2016). After dropping all tildes on rates and concentrations again – this is the notational convention we adopt for the rest of this chapter and for chapter 3 – the time evolution of the concentrations in our expanded network is given by (2.1) for the protein equations, while for the DNA species one has

$$\partial_t x_{(j,\boldsymbol{n})} = \gamma \sum_p \left(k^{p+}_{(j,\boldsymbol{n}-\boldsymbol{e}_p)} x_{(j,\boldsymbol{n}-\boldsymbol{e}_p)} x_p - k^{p+}_{(j,\boldsymbol{n})} x_{(j,\boldsymbol{n})} x_p + k^{p-}_{(j,\boldsymbol{n}+\boldsymbol{e}_p)} x_{(j,\boldsymbol{n}+\boldsymbol{e}_p)} - k^{p-}_{(j,\boldsymbol{n})} x_{(j,\boldsymbol{n})} \right) \tag{2.14}$$

Note that in the above argument we do not require a specific relation between γ and γ'. In reality the rate for producing a single protein from the appropriate DNA, which is of order γ', is lower than the DNA binding/unbinding rates, which are $O(\gamma)$. However, quantifying this would be non-trivial as the model described here simplifies many biological steps into each of the two elementary processes of protein production and binding/unbinding. In any event, for our argument to hold we only need both γ and γ' to be large, and for simplicity we take $\gamma = \gamma'$ as was done in (Rubin and Sollich, 2016).

2.7.2. Obtaining linearised projected equations

We now proceed to use the expanded network to derive linearised projected equations. After the expansion, this requires us to keep track of deviations from steady state of all protein and DNA concentrations. We label these observables by "s" and "b" for subnetwork and bulk proteins as before, and "a" for DNA. Below we will refer to the DNA species in the system as "fast", as expressed mathematically by the factor γ on the r.h.s. of (2.14), while protein species are "slow". Note that the *rate constants* for protein production are also fast, of order γ', but that the resulting *changes* to protein concentrations are slow because of the low concentration of (protein-producing) DNA.

We construct our $\boldsymbol{L}$ matrix by placing the DNA in the bulk, such that $\mathrm{S} = \{\mathrm{s}\}$ and $\mathrm{B} = \{\mathrm{b}, \mathrm{a}\}$, giving the block form

$$\boldsymbol{L} = \left(\begin{array}{c|c} \boldsymbol{L}^{\mathsf{S},\mathsf{S}} & \boldsymbol{L}^{\mathsf{S},\mathsf{B}} \\ \hline \boldsymbol{L}^{\mathsf{B},\mathsf{S}} & \boldsymbol{L}^{\mathsf{B},\mathsf{B}} \end{array}\right) = \left(\begin{array}{c|cc} \boldsymbol{L}^{\mathsf{s},\mathsf{s}} & \boldsymbol{L}^{\mathsf{s},\mathsf{b}} & \boldsymbol{L}^{\mathsf{s},\mathsf{a}} \\ \hline \boldsymbol{L}^{\mathsf{b},\mathsf{s}} & \boldsymbol{L}^{\mathsf{b},\mathsf{b}} & \boldsymbol{L}^{\mathsf{b},\mathsf{a}} \\ \boldsymbol{L}^{\mathsf{a},\mathsf{s}} & \boldsymbol{L}^{\mathsf{a},\mathsf{b}} & \boldsymbol{L}^{\mathsf{a},\mathsf{a}} \end{array}\right) \tag{2.15}$$

The expression (2.7) for the resulting memory function now explicitly involves the fast rate factor γ and the rate constants $\tilde{k}^{p\pm}_{(j,\boldsymbol{n})}$ used in the construction of our expanded network. These parameters appear in the third column in the second expression for $\boldsymbol{L}$ in (2.15), which encodes the time evolution equations for the DNA species. Our remaining task is to take the limit $\gamma \to \infty$, with the aim of obtaining an expression for the memory function that only involves parameters of the original thermodynamic equations (2.1,2.2). To take this limit, one notes that all blocks in the third column of (2.15) are proportional to γ. As the memory function contains an exponential of $\boldsymbol{L}$, it will then have contributions decaying for time differences of order $1/\gamma$, arising from the DNA dynamics, as well as slow contributions decaying on timescales of order unity.

For $\gamma \to \infty$ the fast memory function contributions become effectively instantaneous and add to the rate matrix, and only the slow contributions remain in the memory function. This separation into fast and slow pieces of the memory can be performed using the method in (Rubin and Sollich, 2016) because $\boldsymbol{L}$ has exactly the same division into fast and slow blocks as there. It leads to the simple result that the final rate matrix and (slow) memory function for $\gamma \to \infty$ can be calculated from an effective $\boldsymbol{L}$-matrix no longer involving the fast degrees of freedom. To state the expression for the effective

$\boldsymbol{L}$, we first isolate the slow piece of our original $\boldsymbol{L}$:

$$\boldsymbol{L}_{\backslash a} = \left(\begin{array}{c|c} \boldsymbol{L}^{\mathsf{s,s}} & 0 \\ \hline 0 & \boldsymbol{L}^{\mathsf{b,b}} \end{array}\right) \tag{2.16}$$

(The zero blocks here arise from the fact that subnetwork and bulk proteins do not interact directly in our GRN setting, but only via the DNA species.) The effective $\boldsymbol{L}$-matrix that gives us the rate matrix and memory function for $\gamma \to \infty$ is then of the form $\boldsymbol{L}_{\mathsf{eff}} = \boldsymbol{L}_{\backslash a} + \Delta\boldsymbol{L}_{\backslash a}$, which written out reads:

$$\boldsymbol{L}_{\mathsf{eff}} = \left(\begin{array}{c|c} \boldsymbol{L}^{\mathsf{s,s}} & 0 \\ \hline 0 & \boldsymbol{L}^{\mathsf{b,b}} \end{array}\right) - \left(\begin{array}{c|c} \boldsymbol{L}^{\mathsf{s,a}}(\boldsymbol{L}^{\mathsf{a,a}})^{-1}\boldsymbol{L}^{\mathsf{a,s}} & \boldsymbol{L}^{\mathsf{s,a}}(\boldsymbol{L}^{\mathsf{a,a}})^{-1}\boldsymbol{L}^{\mathsf{a,b}} \\ \hline \boldsymbol{L}^{\mathsf{b,a}}(\boldsymbol{L}^{\mathsf{a,a}})^{-1}\boldsymbol{L}^{\mathsf{a,s}} & \boldsymbol{L}^{\mathsf{b,a}}(\boldsymbol{L}^{\mathsf{a,a}})^{-1}\boldsymbol{L}^{\mathsf{a,b}} \end{array}\right) \tag{2.17}$$

Here the second term including the minus is the additional contribution $\Delta\boldsymbol{L}_{\backslash a}$ from the fast degrees of freedom. The matrix $\boldsymbol{L}_{\mathsf{eff}}$ can be inserted directly into (2.6,2.7) to obtain the projected equations for linearised dynamics, with the fast rate limit already taken.

2.7.3. Fast binding limit as Quasi-Steady State elimination

As in (Rubin and Sollich, 2016), the result (2.17) has a simple interpretation: $\boldsymbol{L}_{\mathsf{eff}}$ can be obtained by eliminating the fast degrees of freedom from the linearised time evolution equations using a QSS condition. Using the block form (2.5), these equations can be written as

$$\partial_t \boldsymbol{\delta x}^{\mathsf{sT}} = \boldsymbol{\delta x}^{\mathsf{sT}}\boldsymbol{L}^{\mathsf{s,s}} + \boldsymbol{\delta x}^{\mathsf{aT}}\boldsymbol{L}^{\mathsf{a,s}} \tag{2.18a}$$

$$\partial_t \boldsymbol{\delta x}^{\mathsf{bT}} = \boldsymbol{\delta x}^{\mathsf{bT}}\boldsymbol{L}^{\mathsf{b,b}} + \boldsymbol{\delta x}^{\mathsf{aT}}\boldsymbol{L}^{\mathsf{a,b}} \tag{2.18b}$$

$$\partial_t \boldsymbol{\delta x}^{\mathsf{aT}} = \boldsymbol{\delta x}^{\mathsf{sT}}\boldsymbol{L}^{\mathsf{s,a}} + \boldsymbol{\delta x}^{\mathsf{bT}}\boldsymbol{L}^{\mathsf{b,a}} + \boldsymbol{\delta x}^{\mathsf{aT}}\boldsymbol{L}^{\mathsf{a,a}} \tag{2.18c}$$

where $\boldsymbol{\delta x}^{\mathsf{sT}}$, $\boldsymbol{\delta x}^{\mathsf{bT}}$ and $\boldsymbol{\delta x}^{\mathsf{aT}}$ are vectors collecting the concentration deviations of sub-network proteins, bulk proteins and DNA species, respectively. All blocks of $\boldsymbol{L}$ appearing in (2.18c) are proportional to γ. This justifies elimination of the fast variables using a QSS condition for these fast (DNA) degrees of freedom, effectively setting the r.h.s. of (2.18c) to zero. The fast species concentrations are then expressed as:

$$\boldsymbol{\delta x}^{\mathsf{aT}} = -\left(\boldsymbol{\delta x}^{\mathsf{sT}}\boldsymbol{L}^{\mathsf{s,a}} + \boldsymbol{\delta x}^{\mathsf{bT}}\boldsymbol{L}^{\mathsf{b,a}}\right)(\boldsymbol{L}^{\mathsf{a,a}})^{-1} \tag{2.19}$$

If we now substitute this back into the time evolution equations (2.18a,2.18b) we obtain:

$$\begin{aligned}
\partial_t \boldsymbol{\delta x}^{\mathsf{sT}} &= \boldsymbol{\delta x}^{\mathsf{sT}}(\boldsymbol{L}^{\mathsf{s,s}} - \underbrace{\boldsymbol{L}^{\mathsf{s,a}}(\boldsymbol{L}^{\mathsf{a,a}})^{-1}\boldsymbol{L}^{\mathsf{a,s}}}_{\Delta\boldsymbol{L}^{\mathsf{s,s}}}) + \boldsymbol{\delta x}^{\mathsf{bT}}(\underbrace{-\boldsymbol{L}^{\mathsf{b,a}}(\boldsymbol{L}^{\mathsf{a,a}})^{-1}\boldsymbol{L}^{\mathsf{a,s}}}_{\Delta\boldsymbol{L}^{\mathsf{b,s}}}) \\
\partial_t \boldsymbol{\delta x}^{\mathsf{bT}} &= \boldsymbol{\delta x}^{\mathsf{sT}}(-\boldsymbol{L}^{\mathsf{s,a}}(\boldsymbol{L}^{\mathsf{a,a}})^{-1}\boldsymbol{L}^{\mathsf{a,b}}) + \boldsymbol{\delta x}^{\mathsf{bT}}(\boldsymbol{L}^{\mathsf{b,b}} - \boldsymbol{L}^{\mathsf{b,a}}(\boldsymbol{L}^{\mathsf{a,a}})^{-1}\boldsymbol{L}^{\mathsf{a,b}})
\end{aligned} \tag{2.20}$$

Writing these time evolution equations in matrix form gives exactly the effective $\boldsymbol{L}$-matrix $\boldsymbol{L}_{\text{eff}}$ defined in (2.17), as claimed. Two terms have been highlighted here by brackets for later comparison.

2.7.4. Equivalence to heuristic linearisation

So far we have shown that the rate matrix and memory functions for the linearised dynamics can be found by applying standard projection results to a set of time evolution equations for only the slow degrees of freedom. This reduced set of equations is obtained by first expanding the thermodynamic equations into a set of mass-action equations, linearising, and finally eliminating the fast variables using a QSS assumption. We next show that this procedure is equivalent to directly linearising the original thermodynamic equations, for any case where there is a timescale separation. This generalises previous work (Rubin and Sollich, 2016) that only considered separate (non-interacting)

fast species. To verify the equivalence, it is useful to write the expanded mass-action equations in the generic form

$$\partial_t x_s = R_s(\boldsymbol{x}^{\mathsf{s}}, \boldsymbol{x}^{\mathsf{b}}, \boldsymbol{x}^{\mathsf{a}}), \quad \partial_t x_b = R_b(\boldsymbol{x}^{\mathsf{s}}, \boldsymbol{x}^{\mathsf{b}}, \boldsymbol{x}^{\mathsf{a}}), \quad \gamma^{-1}\partial_t x_a = R_a(\boldsymbol{x}^{\mathsf{s}}, \boldsymbol{x}^{\mathsf{b}}, \boldsymbol{x}^{\mathsf{a}}) \tag{2.21}$$

where x_s, x_b and x_a are generic components of $\boldsymbol{x}^{\mathsf{s}}$, $\boldsymbol{x}^{\mathsf{b}}$ and $\boldsymbol{x}^{\mathsf{a}}$, respectively. This expanded description is constructed so that for $\gamma \to \infty$, when the x_a can be replaced by their QSS values x_a^*, one recovers the original thermodynamic equations. These can therefore be written as

$$\partial_t x_s = R_s(\boldsymbol{x}^{\mathsf{s}}, \boldsymbol{x}^{\mathsf{b}}, \boldsymbol{x}^{\mathsf{a}*}) \tag{2.22}$$

$$\partial_t x_b = R_b(\boldsymbol{x}^{\mathsf{s}}, \boldsymbol{x}^{\mathsf{b}}, \boldsymbol{x}^{\mathsf{a}*}) \tag{2.23}$$

where the dependence of the x_a^* on $\boldsymbol{x}^{\mathsf{s}}$ and $\boldsymbol{x}^{\mathsf{b}}$ is defined implicitly by

$$R_a(\boldsymbol{x}^{\mathsf{s}}, \boldsymbol{x}^{\mathsf{b}}, \boldsymbol{x}^{\mathsf{a}*}) = 0 \tag{2.24}$$

Expanding now the thermodynamic equations to linear order around a steady state one has

$$\begin{aligned}\partial_t \delta x_s = &\sum_{s'} \frac{\partial R_s}{\partial x_{s'}} \delta x_{s'} + \sum_b \frac{\partial R_s}{\partial x_b} \delta x_b \\ &+ \sum_{s'} \underbrace{\sum_a \frac{\partial x_a^*}{\partial x_{s'}} \frac{\partial R_s}{\partial x_a}}_{\Delta \boldsymbol{L}^{\mathsf{s,s}}} \delta x_{s'} + \sum_b \underbrace{\sum_a \frac{\partial x_a^*}{\partial x_b} \frac{\partial R_s}{\partial x_a}}_{\Delta \boldsymbol{L}^{\mathsf{b,s}}} \delta x_b\end{aligned} \tag{2.25}$$

Here all derivatives are evaluated at the steady state, and the terms in the second line arise from the variation of the QSS values of the fast variables. The required coefficients of the type $\frac{\partial x_a^*}{\partial x_s}$ can be found by differentiating (2.24) with respect to the relevant slow

variable, here x_s:

$$0 = \frac{\partial R_a}{\partial x_s} + \sum_{a'} \frac{\partial R_a}{\partial x_{a'}} \frac{\partial x^*_{a'}}{\partial x_s} \tag{2.26}$$

We can now show that (2.25) is identical to (2.20) derived above. To see this, we note that by linearising the expanded mass action equations (2.21) one can identify the entries of $\boldsymbol{L}$: for example, $\boldsymbol{L}^{\mathsf{b,s}}$ has entries $\partial R_s/\partial x_b$, $\boldsymbol{L}^{\mathsf{s,a}}$ has entries $\gamma\, \partial R_a/\partial x_s$ etc. The QSS coefficients arising from (2.26) can then be written in matrix form as $\partial x^*_a/\partial x_s = -(\boldsymbol{L}^{\mathsf{s,a}}(\boldsymbol{L}^{\mathsf{a,a}})^{-1})_{sa}$, with a similar expression for $\partial x^*_a/\partial x_b$. Inserting these expressions into (2.25) does indeed lead directly to (2.20), with matching terms indicated by the brackets. An exactly analogous argument demonstrates the equivalence for $\partial_t \delta x_b$.

With the above arguments we have justified the heuristic method of Sec. 2.4 for obtaining linearised projected equations for the dynamics of a subnetwork within a larger GRN written in thermodynamic form: expand the GRN equations to linear order in protein concentrations around a steady state. Then construct from this the matrix $\boldsymbol{L}_{\mathsf{eff}}$ and partition this into "s" and "b" blocks according to the chosen subnetwork–bulk split. Finally determine the rate matrix and memory function matrix from (2.6,2.7). We invoked an expanded network of binary reactions involving DNA conformations to *derive* that this is the correct method, but note that its *application* only requires the original thermodynamic equations as input. Therefore the resulting projected equations are, as they should be, independent of the details of the rates $\tilde{k}^{p\pm}_{(j,\boldsymbol{n})}$ used in the construction of the expanded network.

2.7.5. Derivation of nonlinear projected equations

To capture more of the nonlinearity inherent in GRNs one can extend the approach described in the previous sections, to include terms that are quadratic in the deviations from a steady state. The memory function then contains fast and slow contributions, with the former being transformed into additional rate matrix terms when $\gamma \to \infty$. The

resulting rate matrix and slow memory functions can be obtained from an effective $\boldsymbol{L}$-matrix involving only slow (linear and quadratic) observables. As before, this $\boldsymbol{L}_{\text{eff}}$ can be viewed as arising from a QSS elimination of the fast observables. Finally, one can show that this explicit elimination of the fast observables is equivalent to directly expanding the original thermodynamic equations to *quadratic* order in concentration deviations from the steady state: this justifies the heuristic method given in Sec. 2.5.

Except for the last step, the above chain of argument is analogous to the linearised dynamics case, being based entirely on the identical structure of the partition – compare (2.5) and (Sec. 2.8, Eq. 2.27) – of $\boldsymbol{L}$ into fast and slow blocks. The final reduction to a quadratic expansion of the thermodynamic equations is more subtle because the projection approach treats quadratic observables as distinct and not directly tied to products of linear observables. We defer the details to Sec. 2.8 and note only that our derivation there both simplifies and generalises that in (Rubin and Sollich, 2016), from enzyme species that do not interact with each other to arbitrary interactions of the fast degrees of freedom.

2.8. Justification of heuristic method for quadratic observables

In this section we give the details of the projection method with linear and quadratic observables, applied to an expanded network as outlined in Sec. 2.5. The expanded network is one of unary and binary reactions involving the concentrations of subnetwork proteins, $\boldsymbol{x}^{\text{s}}$, of bulk proteins, $\boldsymbol{x}^{\text{b}}$, and of DNA species, $\boldsymbol{x}^{\text{a}}$. The mass action kinetics for this network can be put into the form of an $\boldsymbol{L}$-matrix, defined for linear and quadratic

observables as in (2.10). We partition this into blocks according to

$$\boldsymbol{L} = \left(\begin{array}{c|c} \boldsymbol{L}^{\mathsf{S},\mathsf{S}} & \boldsymbol{L}^{\mathsf{S},\mathsf{B}} \\ \hline \boldsymbol{L}^{\mathsf{B},\mathsf{S}} & \boldsymbol{L}^{\mathsf{B},\mathsf{B}} \end{array}\right) = \left(\begin{array}{c|cc} \boldsymbol{L}^{\tilde{\mathsf{s}},\tilde{\mathsf{s}}} & \boldsymbol{L}^{\tilde{\mathsf{s}},\tilde{\mathfrak{b}}} & \boldsymbol{L}^{\tilde{\mathsf{s}},\tilde{\mathsf{a}}} \\ \hline \boldsymbol{L}^{\tilde{\mathfrak{b}},\tilde{\mathsf{s}}} & \boldsymbol{L}^{\tilde{\mathfrak{b}},\tilde{\mathfrak{b}}} & \boldsymbol{L}^{\tilde{\mathfrak{b}},\tilde{\mathsf{a}}} \\ \boldsymbol{L}^{\tilde{\mathsf{a}},\tilde{\mathsf{s}}} & \boldsymbol{L}^{\tilde{\mathsf{a}},\tilde{\mathfrak{b}}} & \boldsymbol{L}^{\tilde{\mathsf{a}},\tilde{\mathsf{a}}} \end{array}\right) \tag{2.27}$$

Here $\tilde{\mathsf{s}}$ contains the "subnetwork only" observables $\{\mathsf{s}\}$ (linear) and $\{\mathsf{ss}\}$ (quadratic, like $\delta x_s \delta x_{s'}$), while $\{\tilde{\mathfrak{b}}\}$ collects the slow bulk observables $\{\mathsf{b}, \mathsf{sb}, \mathsf{bb}\}$. The fast bulk observables are gathered in $\{\tilde{\mathsf{a}}\}$, which contains $\{\mathsf{a}, \mathsf{sa}, \mathsf{ba}, \mathsf{aa}\}$. Note that with this partitioning of observables we have allocated all fast (DNA) species to the bulk. This is different from the approach in (Rubin and Sollich, 2016) where some fast (enzyme) species were retained in the subnetwork in order to retain more of the nonlinearities. In our case one could similarly keep in the subnetwork those DNA species that produce subnetwork proteins, but it turns out that this makes the final elimination of fast variables rather intricate and so we leave this as an option to pursue in future work.

For our GRN equations, subnetwork and bulk protein species do not interact, so the blocks $\boldsymbol{L}^{\tilde{\mathsf{s}},\tilde{\mathfrak{b}}}$ and $\boldsymbol{L}^{\tilde{\mathfrak{b}},\tilde{\mathsf{s}}}$ are in fact zero. This restriction is not required for our treatment, however, and direct protein-protein interactions could be included in the formalism without modification. As in the case of the linearised dynamics (Sec. 2.7.4), only the third column of (2.27) is fast, *i.e.* has entries proportional to γ (plus subleading terms of order unity arising from the time derivatives of slow-fast product observables such as ba).

From (2.27) one can obtain the rate matrix and memory functions for the projected subnetwork equations, for any finite γ. The limiting values of these quantities for $\gamma \to \infty$ can then be found from a matrix $\boldsymbol{L}_{\mathrm{eff}}$ for only the slow (protein) observables. Our aim in this Appendix is to show that this $\boldsymbol{L}_{\mathrm{eff}}$ is identical to the analogous matrix that one obtains by directly expanding the original slow (GRN) equations to second order in the protein concentrations. This then justifies the heuristic method of constructing the

nonlinear rate matrix and memory functions described in Sec. 2.5.

2.8.1. Generic form of notation

It will be useful to write the full time evolution equations for the expanded network in the generic form

$$\partial_t \boldsymbol{x}^{\mathrm{l}} = \boldsymbol{R}^{\mathrm{l}}(\boldsymbol{x}^{\mathrm{l}}, \boldsymbol{x}^{\mathrm{a}}) \tag{2.28}$$

$$\gamma^{-1}\partial_t \boldsymbol{x}^{\mathrm{a}} = \boldsymbol{R}^{\mathrm{a}}(\boldsymbol{x}^{\mathrm{l}}, \boldsymbol{x}^{\mathrm{a}}) \tag{2.29}$$

These are equations (2.1, 2.14), or more generically (2.21), but we have grouped together the subnetwork and bulk concentration vectors $\boldsymbol{x}^{\mathrm{s}}$ and $\boldsymbol{x}^{\mathrm{b}}$ into a single vector of slow variables $\boldsymbol{x}^{\mathrm{l}}$ to keep the notation for the following discussion compact. The vector $\boldsymbol{x}^{\mathrm{a}}$ contains the fast variables, which in the GRN context are concentrations of DNA conformations, while γ is a fast rate parameter as before. In the limit of large γ, the fast variables are always in QSS with the slow ones so that the expanded network reduces to the thermodynamic state ensemble (in the GRN case) dynamics

$$\partial_t \boldsymbol{x}^{\mathrm{l}} = \boldsymbol{R}^{\mathrm{l}}(\boldsymbol{x}^{\mathrm{l}}, \boldsymbol{x}^{\mathrm{a}}) \tag{2.30}$$

$$0 = \boldsymbol{R}^{\mathrm{a}}(\boldsymbol{x}^{\mathrm{l}}, \boldsymbol{x}^{\mathrm{a}}) \tag{2.31}$$

where the second equation implicitly determines $\boldsymbol{x}^{\mathrm{a}}$ as a function of $\boldsymbol{x}^{\mathrm{l}}$. (In previous sections we marked this QSS value by an asterisk; we omit this here for notational simplicity.) This is the generic form of equations (2.1,2.2).

The matrix $\boldsymbol{L}$ is obtained in the above generic setting by expanding around a fixed

point to second order to write the equations of motion as

$$\partial_t \boldsymbol{x}^{\mathrm{lT}} = \boldsymbol{x}^{\mathrm{lT}}\boldsymbol{L}^{\mathrm{ll}} + \boldsymbol{x}^{\mathrm{aT}}\boldsymbol{L}^{\mathrm{al}} + \boldsymbol{x}^{\mathrm{llT}}\boldsymbol{L}^{\mathrm{ll,l}} + \boldsymbol{x}^{\mathrm{laT}}\boldsymbol{L}^{\mathrm{la,l}} + \boldsymbol{x}^{\mathrm{aaT}}\boldsymbol{L}^{\mathrm{aa,l}} \quad (2.32)$$

$$\gamma^{-1}\partial_t \boldsymbol{x}^{\mathrm{aT}} = \boldsymbol{x}^{\mathrm{lT}}\boldsymbol{L}^{\mathrm{la}} + \boldsymbol{x}^{\mathrm{aT}}\boldsymbol{L}^{\mathrm{aa}} + \boldsymbol{x}^{\mathrm{llT}}\boldsymbol{L}^{\mathrm{ll,a}} + \boldsymbol{x}^{\mathrm{laT}}\boldsymbol{L}^{\mathrm{la,a}} + \boldsymbol{x}^{\mathrm{aaT}}\boldsymbol{L}^{\mathrm{aa,a}} \quad (2.33)$$

All $\boldsymbol{x}$ appearing here and below are deviations $\delta\boldsymbol{x}$ from steady state; we drop the δ to lighten the notation. The $\boldsymbol{x}^{\mathrm{ll}}$ etc are product variables – we assume the indices are ordered to avoid duplicate observables – and the $\boldsymbol{L}$ matrices contain the appropriate derivatives of the "drift" functions R at the fixed point, *e.g.* $\boldsymbol{L}^{\mathrm{ll,a}}$ has elements $L_{ll',a} = \partial_{x_l}\partial_{x_{l'}}R_a$ for $l < l'$ and $L_{ll,a} = (1/2)\partial_{x_l}\partial_{x_l}R_a$ when the two indices are equal. From the above equations then follow the evolution equations for the slow product variables $\boldsymbol{x}^{\mathrm{ll}}$ and the fast products $\boldsymbol{x}^{\mathrm{la}}$ and $\boldsymbol{x}^{\mathrm{aa}}$; see (2.39 & 2.44) below. From the product rule these equations only involve product variables on the r.h.s.; third order terms are in principle present but discarded within the second order expansion. Collecting all variables into a vector $\boldsymbol{z}$ that concatenates $\boldsymbol{x}^{\mathrm{l}}$, $\boldsymbol{x}^{\mathrm{ll}}$, $\boldsymbol{x}^{\mathrm{a}}$, $\boldsymbol{x}^{\mathrm{la}}$, $\boldsymbol{x}^{\mathrm{aa}}$ gives the time evolution equation in the form $\partial_t \boldsymbol{z}^{\mathrm{T}} = \boldsymbol{z}^{\mathrm{T}}\boldsymbol{L}$, where the matrix $\boldsymbol{L}$ has the block form (2.27) if one restores the split of slow observables into subnetwork and bulk.

2.8.2. Heuristic method

The heuristic method involves a direct expansion of the slow equations. To obtain the general form of this, one writes the dynamical equations as

$$\partial_t \boldsymbol{x}^{\mathrm{lT}} = \boldsymbol{x}^{\mathrm{lT}}\boldsymbol{L}^{\mathrm{ll}} + \boldsymbol{x}^{\mathrm{aT}}\boldsymbol{L}^{\mathrm{al}} + (\boldsymbol{x}^{\mathrm{l}} \circ \boldsymbol{x}^{\mathrm{l}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{ll,l}} + (\boldsymbol{x}^{\mathrm{l}} \circ \boldsymbol{x}^{\mathrm{a}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{la,l}} + (\boldsymbol{x}^{\mathrm{a}} \circ \boldsymbol{x}^{\mathrm{a}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{aa,l}} \quad (2.34)$$

$$\gamma^{-1}\partial_t \boldsymbol{x}^{\mathrm{aT}} = \boldsymbol{x}^{\mathrm{lT}}\boldsymbol{L}^{\mathrm{la}} + \boldsymbol{x}^{\mathrm{aT}}\boldsymbol{L}^{\mathrm{aa}} + (\boldsymbol{x}^{\mathrm{l}} \circ \boldsymbol{x}^{\mathrm{l}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{ll,a}} + (\boldsymbol{x}^{\mathrm{l}} \circ \boldsymbol{x}^{\mathrm{a}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{la,a}} + (\boldsymbol{x}^{\mathrm{a}} \circ \boldsymbol{x}^{\mathrm{a}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{aa,a}} \quad (2.35)$$

where the "circle" product indicates the actual products of the regular linear observables, with the same index ordering as in the projection approach (so that all the $\boldsymbol{L}$-matrices are as before). One now needs to determine $\boldsymbol{x}^{\mathrm{a}}$ by setting the r.h.s. of (2.35) to zero, and substitute the result into (2.34). As we are only expanding to second order in $\boldsymbol{x}^{\mathrm{l}}$, it is enough also to obtain $\boldsymbol{x}^{\mathrm{a}}$ to this order. Starting with the first order of (2.35) one obtains

$$\boldsymbol{x}^{\mathrm{a}\mathrm{T}} = -\boldsymbol{x}^{\mathrm{l}\mathrm{T}}\boldsymbol{L}^{\mathrm{la}}(\boldsymbol{L}^{\mathrm{aa}})^{-1} \equiv \boldsymbol{x}_0^{\mathrm{a}\mathrm{T}} \tag{2.36}$$

where $\boldsymbol{x}_0^{\mathrm{a}}$ will be a convenient shorthand. (Note that the coefficient matrix $-\boldsymbol{L}^{\mathrm{la}}(\boldsymbol{L}^{\mathrm{aa}})^{-1}$ is the one we worked out below (2.26), in a slightly more pedestrian fashion.) All *second* order terms in (2.35) can now be evaluated to the required accuracy by replacing $\boldsymbol{x}^{\mathrm{a}}$ by $\boldsymbol{x}_0^{\mathrm{a}}$. Solving for $\boldsymbol{x}^{\mathrm{a}}$ then gives

$$\boldsymbol{x}^{\mathrm{a}\mathrm{T}} = \boldsymbol{x}_0^{\mathrm{a}\mathrm{T}} - \left[(\boldsymbol{x}^{\mathrm{l}} \circ \boldsymbol{x}^{\mathrm{l}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{ll,a}} + (\boldsymbol{x}^{\mathrm{l}} \circ \boldsymbol{x}_0^{\mathrm{a}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{la,a}} + (\boldsymbol{x}_0^{\mathrm{a}} \circ \boldsymbol{x}_0^{\mathrm{a}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{aa,a}}\right](\boldsymbol{L}^{\mathrm{aa}})^{-1} \tag{2.37}$$

Inserting into (2.34) and dropping terms of higher than quadratic order gives the required expansion of the slow equations,

$$\begin{aligned}\partial_t \boldsymbol{x}^{\mathrm{l}\mathrm{T}} =& \boldsymbol{x}^{\mathrm{l}\mathrm{T}}\boldsymbol{L}^{\mathrm{ll}} + \boldsymbol{x}_0^{\mathrm{a}\mathrm{T}}\boldsymbol{L}^{\mathrm{al}} + (\boldsymbol{x}^{\mathrm{l}} \circ \boldsymbol{x}^{\mathrm{l}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{ll,l}} + (\boldsymbol{x}^{\mathrm{l}} \circ \boldsymbol{x}_0^{\mathrm{a}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{la,l}} + (\boldsymbol{x}_0^{\mathrm{a}} \circ \boldsymbol{x}_0^{\mathrm{a}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{aa,l}} \\ &- \left[(\boldsymbol{x}^{\mathrm{l}} \circ \boldsymbol{x}^{\mathrm{l}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{ll,a}} + (\boldsymbol{x}^{\mathrm{l}} \circ \boldsymbol{x}_0^{\mathrm{a}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{la,a}} + (\boldsymbol{x}_0^{\mathrm{a}} \circ \boldsymbol{x}_0^{\mathrm{a}})^{\mathrm{T}}\boldsymbol{L}^{\mathrm{aa,a}}\right](\boldsymbol{L}^{\mathrm{aa}})^{-1}\boldsymbol{L}^{\mathrm{al}}\end{aligned} \tag{2.38}$$

This then determines the $\boldsymbol{L}$-matrix for the slow variables from which rate matrix and memory functions are determined in the heuristic approach.

2.8.3. Expanded network approach

As explained before, based on the reasoning in (Rubin and Sollich, 2016), if one writes down expressions for the rate matrix and memory functions from the full matrix $\boldsymbol{L}$ and

then takes the fast rate limit $\gamma \to \infty$, the resulting rate matrix and (slow) memory function can be found from a matrix $\boldsymbol{L}_{\rm eff}$ describing only the dynamics of the slow variables $\boldsymbol{x}^{\rm l}$ and $\boldsymbol{x}^{\rm ll}$. This $\boldsymbol{L}_{\rm eff}$ is obtained by eliminating the fast variables $\boldsymbol{x}^{\rm a}$, $\boldsymbol{x}^{\rm la}$ and $\boldsymbol{x}^{\rm aa}$ using the condition that they are in QSS. These conditions are, within the projection approach, *linear* equations because product observables are treated as independent from linear observables.

What we then need to demonstrate is that this distinct elimination assigns to $\boldsymbol{x}^{\rm la}$ and $\boldsymbol{x}^{\rm aa}$ the same values as the direct expansion approach, namely $(\boldsymbol{x}^{\rm l} \circ \boldsymbol{x}^{\rm a}_0)$ and $(\boldsymbol{x}^{\rm a}_0 \circ \boldsymbol{x}^{\rm a}_0)$. Once this is shown, it follows that the linear fast variables $\boldsymbol{x}^{\rm a}$ are eliminated in the same way in the two approaches, because the same quadratic fast variables are substituted into the relevant equations (2.33 & 2.35). Thus *all* fast variables are eliminated in the same way from the time evolution equation for the slow variables, to the quadratic order we consider here. As the slow $\boldsymbol{L}$-matrix in the heuristic approach and the $\boldsymbol{L}_{\rm eff}$ in the expanded network method are both obtained from this time evolution equation for the slow variables, they are therefore equal as we want to show.

Elimination of $\boldsymbol{x}^{\rm la}$

Using the product rule of differentiation, the equations of motion for the $\boldsymbol{x}^{\rm la}$ are

$$\partial_t \boldsymbol{x}^{\rm laT} = \gamma((\boldsymbol{x}^{\rm lT} \circ \boldsymbol{x}^{\rm lT}\boldsymbol{L}^{\rm la})) + \gamma((\boldsymbol{x}^{\rm lT} \circ \boldsymbol{x}^{\rm aT}\boldsymbol{L}^{\rm aa})) + ((\boldsymbol{x}^{\rm lT}\boldsymbol{L}^{\rm ll} \circ \boldsymbol{x}^{\rm aT})) + ((\boldsymbol{x}^{\rm aT}\boldsymbol{L}^{\rm al} \circ \boldsymbol{x}^{\rm aT})) \tag{2.39}$$

Here the double brackets on the right indicate that after the circle products are taken the actual products have to be replaced by product variables, to remain within the projection framework.

We want to eliminate the $\boldsymbol{x}^{\rm la}$ from the condition that the r.h.s. vanishes. Fortunately for large γ the first two terms, which stem from the time evolution of $\boldsymbol{x}^{\rm a}$, dominate; the

last two become negligible in comparison. Thus one has to solve

$$((\boldsymbol{x}^{\mathrm{lT}} \circ \boldsymbol{x}^{\mathrm{lT}} \boldsymbol{L}^{\mathrm{la}})) + ((\boldsymbol{x}^{\mathrm{lT}} \circ \boldsymbol{x}^{\mathrm{aT}} \boldsymbol{L}^{\mathrm{aa}})) = 0 \tag{2.40}$$

From the structure of this one sees that the $\boldsymbol{x}^{\mathrm{l}}$ only act as "spectators", while considering the second factors one has to solve the same equation as at linear order. The solution is therefore expected to be $\boldsymbol{x}^{\mathrm{laT}} = ((\boldsymbol{x}^{\mathrm{lT}} \circ \boldsymbol{x}_0^{\mathrm{aT}}))$ as we want to show.

To see this in more detail we write out (2.40) in components:

$$\sum_{l'} x_{(ll')} L_{l'a} + \sum_{a'} x_{la'} L_{a'a} = 0 \tag{2.41}$$

Here we have written $x_{(ll')}$ to indicate that the indices are to be taken as ordered, *i.e.* $x_{(ll')} = x_{ll'}$ if $l \leq l'$ and $x_{(ll')} = x_{l'l}$ otherwise. The proposed solution is

$$x_{la} = ((\boldsymbol{x}^{\mathrm{lT}} \circ \boldsymbol{x}_0^{\mathrm{aT}}))_{la} = -\sum_{l'a'} x_{(ll')} L_{l'a'} (\boldsymbol{L}^{\mathrm{aa}})^{-1}_{a'a} \tag{2.42}$$

This does solve (2.41) because

$$\sum_{a''} x_{la''} L_{a''a} = -\sum_{l'a'a''} x_{(ll')} L_{l'a'} (\boldsymbol{L}^{\mathrm{aa}})^{-1}_{a'a''} L_{a''a} = -\sum_{l'} x_{(ll')} L_{l'a} \tag{2.43}$$

2.8.4. Elimination of x^{aa}

We proceed again using the product rule of differentiation, which gives as the equations of motion for the $\boldsymbol{x}^{\mathrm{aa}}$

$$\partial_t \boldsymbol{x}^{\mathrm{aaT}} = \gamma((\boldsymbol{x}^{\mathrm{lT}} \boldsymbol{L}^{\mathrm{la}} \circ \boldsymbol{x}^{\mathrm{aT}})) + \gamma((\boldsymbol{x}^{\mathrm{aT}} \boldsymbol{L}^{\mathrm{aa}} \circ \boldsymbol{x}^{\mathrm{aT}})) + \gamma((\boldsymbol{x}^{\mathrm{aT}} \circ \boldsymbol{x}^{\mathrm{lT}} \boldsymbol{L}^{\mathrm{la}})) + \gamma((\boldsymbol{x}^{\mathrm{aT}} \circ \boldsymbol{x}^{\mathrm{aT}} \boldsymbol{L}^{\mathrm{aa}})) \tag{2.44}$$

Here all terms contribute for large γ, but one sees that in the first two the right factor of $\boldsymbol{x}^{\mathrm{a}}$ is again a "spectator" and similarly with the left factor for the last two terms. Accordingly one can show that the proposed solution, which is $\boldsymbol{x}^{\mathrm{aa\,T}} = ((\boldsymbol{x}_0^{\mathrm{a\,T}} \circ \boldsymbol{x}_0^{\mathrm{a\,T}}))$, ensures that each pair of terms vanishes separately.

For explicit calculation it is again useful to write out components. The aa' component of the first two terms of (2.44), without the overall factor of γ, reads

$$\sum_{l'} x_{l'a'} L_{l'a} + \sum_{a''} x_{(a'a'')} L_{a''a} \tag{2.45}$$

The proposed solution is

$$x_{aa'} = ((\boldsymbol{x}_0^{\mathrm{a\,T}} \circ \boldsymbol{x}_0^{\mathrm{a\,T}}))_{aa'} = \sum_{l_1 l_2 a_1 a_2} x_{(l_1 l_2)} L_{l_1 a_1} (\boldsymbol{L}^{\mathrm{aa}})^{-1}_{a_1 a} L_{l_2 a_2} (\boldsymbol{L}^{\mathrm{aa}})^{-1}_{a_2 a'} \tag{2.46}$$

Substituting this and the solution (2.42) for x_{la} turns (2.45) into

$$-\sum_{l' l'' a''} x_{(l' l'')} L_{l'' a''} (\boldsymbol{L}^{\mathrm{aa}})^{-1}_{a'' a'} L_{l' a} + \sum_{a'' l_1 l_2 a_1 a_2} x_{(l_1 l_2)} L_{l_1 a_1} (\boldsymbol{L}^{\mathrm{aa}})^{-1}_{a_1 a'} L_{l_2 a_2} (\boldsymbol{L}^{\mathrm{aa}})^{-1}_{a_2 a''} L_{a'' a} \tag{2.47}$$

The last factors in the second term again cancel, reducing it to

$$\sum_{l_1 l_2 a_1} x_{(l_1 l_2)} L_{l_1 a_1} (\boldsymbol{L}^{\mathrm{aa}})^{-1}_{a_1 a'} L_{l_2 a} \tag{2.48}$$

After a relabelling of summation indices this is identical to the first term. This means that (2.45) vanishes, *i.e.* the first two terms on the r.h.s. of (2.44) cancel. Similarly the last two terms vanish, showing that (2.46) is the correct QSS assignment of the $\boldsymbol{x}^{\mathrm{aa}}$.

Equations of motion for $\boldsymbol{x}^{\mathrm{ll}}$

Above we have shown that the direct and the projection elimination procedures give the same time evolution equation for $\boldsymbol{x}^{\mathrm{l}}$. The same can then also be checked straightfor-

wardly for the product variables $(\boldsymbol{x}^{\mathrm{l}} \circ \boldsymbol{x}^{\mathrm{l}})$ and their projection analogues $\boldsymbol{x}^{\mathrm{ll}}$. By analogy with (2.39), the latter evolve in time according to

$$\partial_t \boldsymbol{x}^{\mathrm{llT}} = ((\boldsymbol{x}^{\mathrm{lT}} \circ \boldsymbol{x}^{\mathrm{lT}} \boldsymbol{L}^{\mathrm{ll}})) + ((\boldsymbol{x}^{\mathrm{lT}} \circ \boldsymbol{x}^{\mathrm{aT}} \boldsymbol{L}^{\mathrm{al}})) + ((\boldsymbol{x}^{\mathrm{lT}} \boldsymbol{L}^{\mathrm{ll}} \circ \boldsymbol{x}^{\mathrm{lT}})) + ((\boldsymbol{x}^{\mathrm{aT}} \boldsymbol{L}^{\mathrm{al}} \circ \boldsymbol{x}^{\mathrm{lT}})) \quad (2.49)$$

The actual products obey the same equation, just written differently:

$$\partial_t (\boldsymbol{x}^{\mathrm{l}} \circ \boldsymbol{x}^{\mathrm{l}})^{\mathrm{T}} = (\boldsymbol{x}^{\mathrm{lT}} \circ \boldsymbol{x}^{\mathrm{lT}} \boldsymbol{L}^{\mathrm{ll}}) + (\boldsymbol{x}^{\mathrm{lT}} \circ \boldsymbol{x}^{\mathrm{aT}} \boldsymbol{L}^{\mathrm{al}}) + (\boldsymbol{x}^{\mathrm{lT}} \boldsymbol{L}^{\mathrm{ll}} \circ \boldsymbol{x}^{\mathrm{lT}}) + (\boldsymbol{x}^{\mathrm{aT}} \boldsymbol{L}^{\mathrm{al}} \circ \boldsymbol{x}^{\mathrm{lT}}) \quad (2.50)$$

From both, the fast products $(\boldsymbol{x}^{\mathrm{l}} \circ \boldsymbol{x}^{\mathrm{a}})$ (respectively $\boldsymbol{x}^{\mathrm{la}}$) and $(\boldsymbol{x}^{\mathrm{a}} \circ \boldsymbol{x}^{\mathrm{a}})$ (respectively $\boldsymbol{x}^{\mathrm{aa}}$) then need to be eliminated. As we have already established that these eliminations are identical, also the resulting equations for the ll-product variables must be identical.

2.9. Summary

In summary we have developed a method using Zwanzig-Mori projections that is generally applicable to GRNs and allows the extraction of information in the form of memory functions. The subnetwork can be defined arbitrarily and the type of insight that memory functions provides will depend on the choice of such subnetworks.

In our mathematical treatment we took advantage of an argument of timescale separation for molecular mechanisms operating in transcriptional networks, in which fast processes (DNA-protein interactions) are assumed to be in QSS compared to the slow processes (changes in protein concentration). We also assume the effective concentration of DNA species to be small in comparison to that of protein species. We consider these to be reasonable assumptions because DNA-protein interactions occur at a much faster rate than the changes in protein concentrations produced by transcription and translation (Garcia et al., 2010). Moreover, there are usually at most four DNA copies of each gene per cell, whereas the number of individual protein molecules is normally

considerably higher. The production of individual proteins per DNA copy number, *i.e.* the relevant rate constant, can then be large while the resulting relative changes to protein concentration are slow. These assumptions allowed us to construct an extended reaction network involving fast molecular (DNA) species and at most binary reactions. This construction is sufficiently general for it to be applicable to other systems with a timescale separation, as long as they satisfy the above conditions for the relevant concentrations and rate constants.

An important component of this Zwanzig-Mori projection method is that the memory functions describe all the behaviours of a system in the vicinity of a steady state. Even though this comes at the cost of predictions becoming increasingly inaccurate the further the system is from the fixed point, as we will show in Chapter 3, the information around the steady state is sufficient to distil relevant properties of the network. We therefore employ the method as a tool for probing the dynamics near a chosen steady state, rather than for tracking dynamics. The benefit is that the resulting memory functions are substantially simpler than the original system. There is no need to perform simulations to analyse memory functions or to test different positions in phase space as a steady state is approached. Moreover, the method can also be applied to an arbitrarily chosen subnetwork of a generic network whereas other model reduction approaches – as discussed in the Introduction – do not allow this, and are also more restrictive as they require specific network motifs or functional forms of the dynamics.

The memory functions produced by the Zwanzig-Mori approach allow exploration of the amplitudes and timescales of the interactions between the subnetwork and bulk. This can reveal information that is not otherwise easy to discern from the parameters of the full dynamics or from the functional form of the equations. Each memory function represents the total of the contributions from a given subnetwork component that feeds back to the subnetwork over time after passing through the bulk. This total memory

can be decomposed into channels that describe the flow of signals through specific components in the bulk. In this way, the channels that dominate a memory effect can be identified, providing insight into the dynamical mechanisms responsible for achieving and maintaining a steady state. In Chapter 3 we exemplify how memory functions help to extract features of the dynamics that would not be readily detectable in direct simulations of the time evolution equations.

3

Case studies around a fixed point

In this chapter we demonstrate the applications of the approach described in Chapter 2 with case studies. We apply the projection by choosing subnetworks aided by our intuition of each of these systems. We first illustrate the memory function method on a simple biologically relevant example system: a cross-repression motif. We then apply the method to a system directly modelling biology: A gene regulatory network that directs ventral neural tube patterning (Sec. 3.2). We demonstrate the advantages of this method and what kind of insights it can provide.

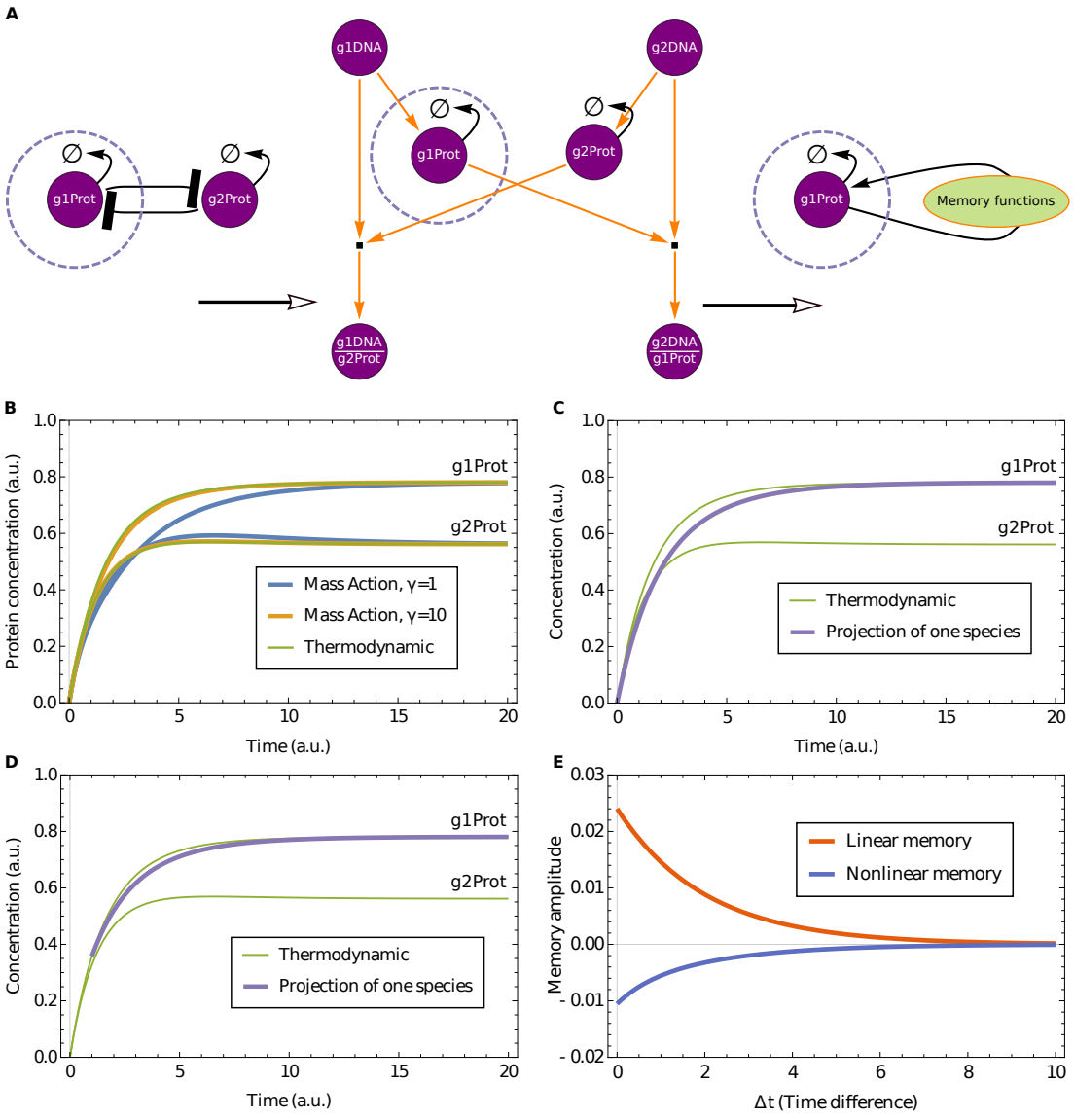

A
g1DNA
g2DNA
g1Prot
g2Prot
Memory functions
g1DNA g2Prot
g2DNA g1Prot
B
Protein concentration (a.u.)
Time (a.u.)
g1Prot
g2Prot
Mass Action, γ=1
Mass Action, γ=10
Thermodynamic
C
Concentration (a.u.)
Time (a.u.)
Thermodynamic
Projection of one species
D
Concentration (a.u.)
Time (a.u.)
E
Memory amplitude
Δt (Time difference)
Linear memory
Nonlinear memory

Figure 3.1: Example of application of Zwanzig-Mori projection. (**A**) Illustration of the methodology, for the example of a cross repressive motif. First the nonlinear thermodynamic reactions (left) are expanded into mass action reactions with an appropriate timescale separation (centre). This generates additional nodes that represent the possible DNA conformations for both proteins, with *e.g.* g1DNA/g2Prot indicating DNA for gene 1 with protein 2 bound to it. To the expanded network we can apply the projection approach, retaining only the concentration of protein 1 in the subnetwork. The effect of the rest of the network – the bulk – is captured via memory terms (right). (**B**) Comparison of the cross repressive motif described via the original thermodynamic equations and the expanded mass action equations with and without timescale separation. Already for a moderate fast rate factor of $\gamma = 10$ the mass action and thermodynamic time evolutions are visually indistinguishable. Two time courses are plotted in Fig. (**B**-**D**), for g1Prot & g2Prot. (**C**-**D**) Demonstration of the projection approach with g1Prot in subnetwork and g2Prot in bulk. The projected equations track the dynamics of the original thermodynamic equation with a reduced system that contains memory functions. However, one can observe in (**C**) that accuracy can be lost in the transient if the system is initiated too far away from the fixed steady state. (**D**) Starting nearer to the steady state (at $t = 1$) substantially increases the accuracy of the projected description. (**E**) Example of memory functions of protein 1 to itself, decaying with time difference. The linear memory function is positive as expected from the network (positive feedback loop), while the nonlinear term is negative to correct for range-limiting nonlinearities that the linear terms cannot capture. Parameters used are $\alpha_1 = 1$, $\alpha_2 = 1$, $w_{\mathsf{p1}} = 1$, $w_{\mathsf{p2}} = 2$, $w_1 = 2$, $w_2 = 2$, $\beta_1 = 1/2$, $\beta_2 = 1/2$.

3.1. Zwangzig-Mori projection of a cross-repression motif

We applied the Zwanzig-Mori projection method to a cross repressive motif to illustrate our approach. Fig. 3.1A shows the expansion procedure for this very simple regulation network: each of two genes is repressed through one binding site for the other factor. This system can be expanded to represent all possible DNA conformations, bound and unbound, as shown graphically by the four outer nodes in the network in the centre of Fig. 3.1A. As the arrows indicate, in the expanded network unbound DNA can bind protein to form bound DNA; the reverse process is also possible. In this way one creates a set of reaction equations that has been extended and brought into mass action form by including the concentration of DNA conformations. The equations used for Fig. 3.1

are of the form of (2.1) & ()2.2):

$$\partial_t[\text{g1Prot}] = \alpha_1 \left(w_1 + w_{\text{p2}}[\text{g2Prot}]\right)^{-1} - \beta_1[\text{g1Prot}] \tag{3.1}$$

and conversely for g2Prot. In numerical simulations of the expanded network we first checked the fast rate limit, which we control by setting the ratio, γ, of typical DNA reaction rates to protein production rates. Fig. 3.1B shows that already with a moderate value of $\gamma = 10$ the predicted time courses of protein concentrations are almost identical to those predicted by the thermodynamic equations. As $\gamma \to \infty$ the trajectories from the mass action equations and the thermodynamic equations become identical.

We chose to separate the system so that one protein species, g1Prot, is in the subnetwork and the other, g2Prot, is in the bulk. After expanding into protein and DNA species we proceeded to place all DNA species along with g2Prot in the bulk. Because we have performed the expansion around a steady state, the method is more precise when close to the steady state as expected (Fig. 3.1D). Note that the cross-repressive motif with one binding site is not bistable. It could be made bistable by adding binding sites; with the current approach the behaviour around each steady state would then have to be analysed separately, as we also do in the neural tube application in the Sec. 3.2. More general projection methods can overcome this limitation as detailed in the Discussion, although it is clear that there are limits of principle here. For example, where the dynamical choice of steady state depends on the initial concentration of a bulk species, this effect cannot be captured by a subnetwork description that by definition does not track this species.

Next we analyse the memory functions in our cross-repressive example, which can be thought of as describing the strength of a signal returning from the bulk a specific time after the original signal entered the bulk from the subnetwork (Fig. 3.1E). The amplitude of the leading *linear* memory function – where the signal is $\delta x_1(t')$ – is positive. This

makes sense as, in the cross repressive motif, a protein is inhibiting its inhibitor, thus promoting its own production. The second order memory function governs the effects of the signal $\delta x_1^2(t')$. It is, in this case, negative and acts as a correction that captures nonlinearities beyond the linear memory. We observe that the decay in time of the memory functions is determined by the decay rate of the bulk protein. This is consistent with the memory function describing a protein repressing its own repressor, hence the effect only lasts as long as the repressor protein is present. Overall, the memory functions provide compact and readable descriptions of how the bulk modifies and influences the activity of the subnetwork.

3.2. Application to neural tube network

Reducing part of a network into memory functions offers a way to simplify and analyse the effect of the factors in this bulk part on the remaining subnetwork. To illustrate this approach we chose to analyse a four gene network involved in the embryonic patterning of the vertebrate neural tube (Cohen et al., 2013, Dessaud et al., 2008). The developing neural tube is a well characterised example of developmental pattern formation. In this tissue, the secreted molecule, Sonic-Hedgehog (Shh) forms a ventral to dorsal gradient (Roelink et al., 1995). This in turn generates discrete domains of gene expression that define the progenitors of the distinct neuronal subtypes that comprise spinal cord circuitry (Fig. 3.2A,B). A transcriptional network has been identified that is controlled by graded Shh signaling and is responsible for specifying ventral progenitor domains (Briscoe et al., 2000, Cohen et al., 2013). Genetic and molecular experiments have determined the activity of the components of this network (Fig. 3.2C) and documented the temporal dynamics of pattern formation in vitro and in vivo (Balaskas et al., 2012, Cohen et al., 2013, 2014). For the three most ventral domains of progenitors, four transcription factors are essential, Nkx2.2, Olig2, Pax6 and Irx3. The most ventral domain (p3 domain) is

characterised by high Nkx2.2 and low levels of the other three proteins; the domain adjacent to this (pMN domain) has high levels of Olig2, medium levels of Pax6 and low levels of the other two proteins. Dorsal to this, the p2 domain expresses high levels of Pax6 and Irx3 and low levels of the other two proteins.

Mathematical models have been formulated based on the experimental data and these are able to replicate key aspects of cell patterning (Balaskas et al., 2012, Cohen et al., 2014) (Fig. 3.2D). We take advantage of these to develop a Zwanzig-Mori projection of the system (using equations as described in (Cohen et al., 2014) & A.2). To this end, we chose Nkx2.2 and Olig2 to be the subnetwork species, given that they are receiving direct input from Shh (Fig. 3.2C), and replaced Irx3 and Pax6 with memory functions.

3.2.1. Linear memory analysis

We first examined the properties of the linear memory functions. These are the most substantial contributions, close to the specific steady states, of the species in the bulk (Pax6 and Irx3) on the temporal changes in the activity of Nkx2.2 and Olig2. We note that the system is multistable (Fig. 3.2E) and different combinations of steady states are available at different positions along the dorsal ventral axis (Shh gradient). We therefore analysed each possible steady state along the neural tube. In this case, as the bulk species are mutual repressors with the subnetwork species, they form a positive feedback loop with the factors in the subnetwork. As a consequence, each of the linear memory functions is positive and exponentially decaying on the time-scale of protein degradation of the bulk species. Thus the relative contribution of each of the memory functions can be assessed by comparing their amplitude, defined as the memory function value at zero time difference.

We calculated the amplitudes of the memory functions and determined the memory

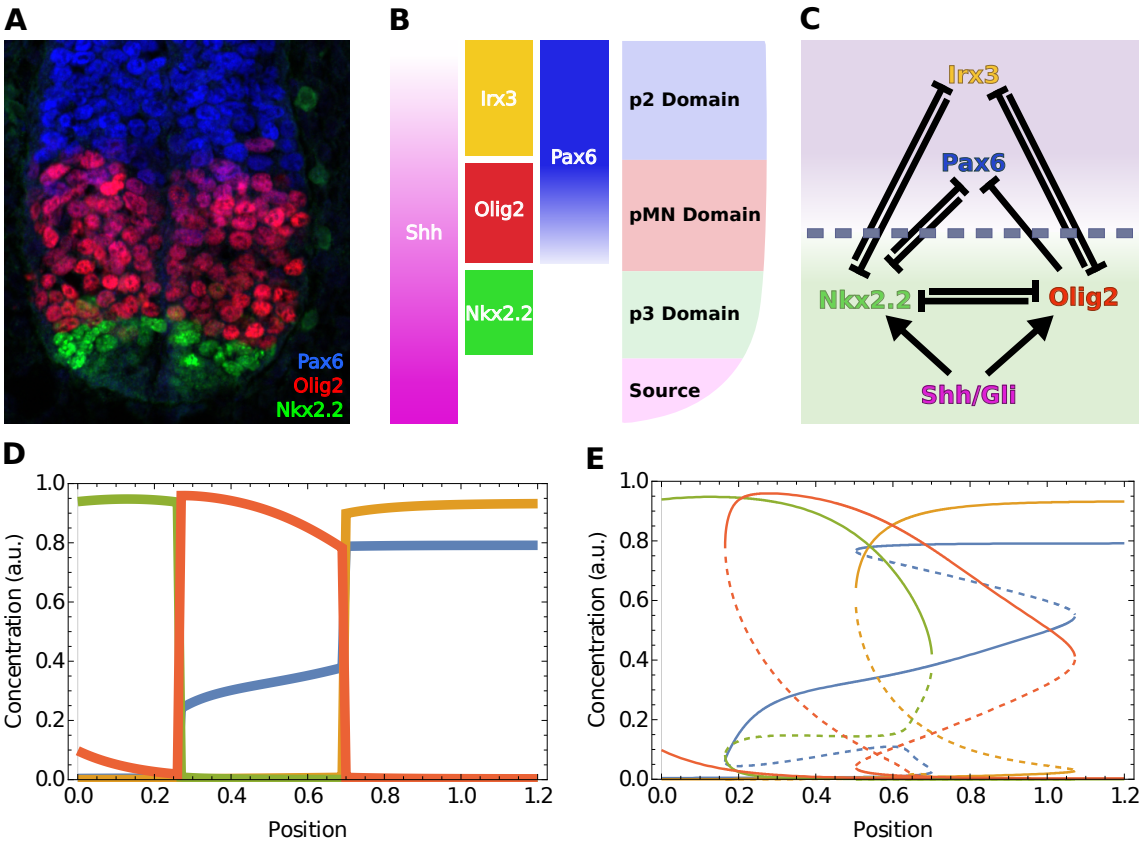


Figure 3.2: Patterning of the vertebrate neural tube. (**A**) Antibody staining of Wild type (WT) mouse stained for three of the main bands in dorso-ventral patterning. (Image provided by Katherine Exelby). (**B**) Illustration of neural tube patterning: ventral Shh secreted from the notocord and floor plate (termed "Source") generates patterned domains along the dorso-ventral axis. Each domain is defined by the expression of a characteristic set of genes. (**C**) GRN that patterns the three most ventral domains of the neural tube. The chosen separation of bulk (purple) and subnetwork (green) in the application of the Zwanzig-Mori projection is also shown. (**D**) Simulations of steady state pattern along the dorsoventral axis using a set of thermodynamic equations of the form of equations (2.1,2.2). These equations were taken – along with appropriate initial conditions of 0 for all species – from (Cohen et al., 2014) (see Appendix A.1 for details). For all plots where x axis represents neural tube position, zero corresponds to the most ventral point. (**E**) Full bifurcation diagram illustrating the multistable nature of the network. Shown are steady state concentrations of the four molecular species against neural tube position, with unstable steady states marked dashed. Colours in (**D**, **E**) identify genes/proteins in the same way as in the labelling of the illustration (**B**) and of the network nodes in (**C**). This colour code is used throughout this chapter unless otherwise noted.

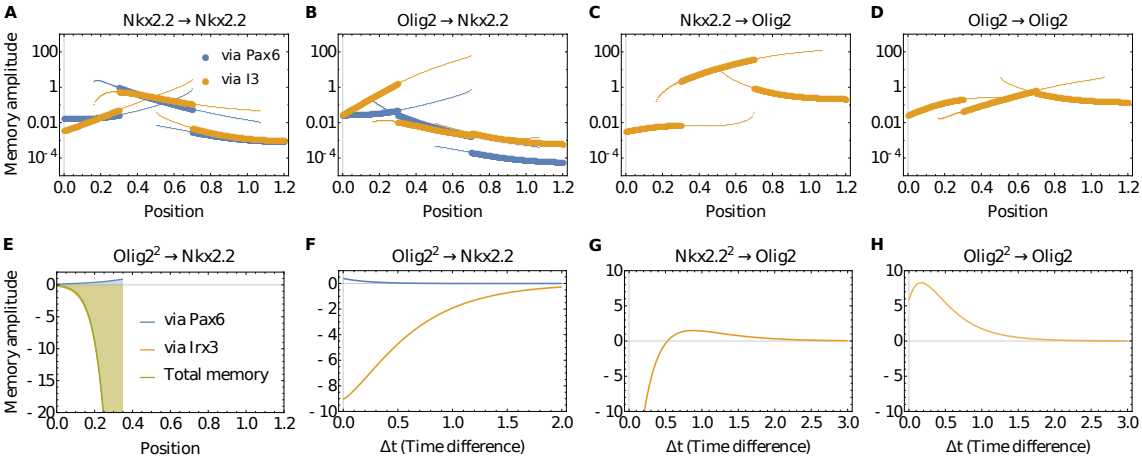


Figure 3.3: Memory amplitude and temporal dynamics. (**A**) Amplitude of memory (memory function at $\Delta t = 0$) of Nkx2.2 to itself along the neural tube. There are multiple lines as the analysis was performed at all possible stable steady states. The vertical axis is logarithmic to make the range of amplitudes easier to appreciate. Colours identify the memory amplitude contribution from each of the two possible bulk channels, via Irx3 and Pax6, respectively. Thick lines indicate physiological states, while thin lines indicate states that are not usually observed *in vivo*. (**B**) Linear memory amplitude of (past) Olig2 on Nkx2.2 along the neural tube. The memory via Pax6 is for the most part below the memory via Irx3 in each pair of corresponding curves. (**C,D**) Memory amplitudes of Olig2 to Nkx2.2 (**C**) and to itself (**D**). No channel decomposition is performed as Olig2 receives memory only via the Irx3 channel. (**E**) Nonlinear memory of (past) Olig2 squared on Nkx2.2 in the p3 domain, where dynamics are dominated by Irx3; memory via Pax6 is negligible by comparison. (**F**) Nonlinear memory function of (past) Olig2 squared on Nkx2.2 from position 0.2 in (**E**), plotted to show that the relative contribution of the Pax6 channel is small also for all time differences. (**G,H**) Nonlinear memory functions of Olig2 to Nkx2.2 (**G**) and itself (**H**) in the p2 domain at position 0.7, exemplifying the potential for nontrivial time dependences (including non-monotonicity and sign changes) in the nonlinear memory functions.

effects that Nkx2.2 and Olig2 receive over time. Furthermore, we used the method developed in (Rubin et al., 2014) to *decompose* the memory into contributions from memory signals passing through the two different bulk species (Fig. 3.3A–D). Briefly, this consists in writing out the $\boldsymbol{L}^{\mathrm{S,B}}$ and $\boldsymbol{L}^{\mathrm{B,S}}$ matrices as sums which can be analysed individually. This allowed us to determine the importance of specific regulatory interactions between transcription factors (TFs) at every position of the neural tube model, and assess their respective contributions. The results indicate that, for the most part, there is only a small memory amplitude of Nkx2.2 to the past of Olig2 passing through

Pax6 (Fig. 3.3B; note the logarithmic y-axis): the memory of (past) Olig2 on Nkx2.2 is largely dominated by memory through Irx3. The only exception is for steady states that are not reached during normal neural tube patterning (lower pair of curves in the neural tube position range $\approx [0.2 \dots 0.45]$ in Fig. 3.3B). These steady states exist because the system is multistable, but the initial conditions that lead to them are incompatible with physiological conditions. We additionally observe in Fig. 3.3A that the memory of Nkx2.2 to itself is for the most part shared between Irx3 and Pax6, with no particular TF being dominant. There are again exceptions from this, but only in steady states that are biologically unreasonable. In the case of the memory of Olig2, it can only be influenced by Irx3 as is clear from the structure of the network in Fig. 3.2C. Channel decomposition is therefore unnecessary. We observe that the memory effects of Olig2 to itself are strongest in the p3 domain, and the opposite is true (memory of Nkx2.2 on Olig2 is stronger) in the pMN domain

We observe in Fig. 3.3 that the order of magnitude of the memory amplitudes changes substantially with neural tube position. This is a consequence of the system becoming more or less sensitive to concentration fluctuations of a given species within the network. An example of this is Nkx2.2 memory to itself via Pax6 in the pMN domain (Fig. 3.3A), where dorsally the lower levels of Shh mean that the binding sites of Nkx2.2 are less occupied by the active form of Gli (Gli is the transcriptional effector downstream of Shh that binds to Nkx2.2 & Olig2, see (Cohen et al., 2014) for details on how this is implemented). Active Gli sets the rate of production of Nkx2.2 and thus has a direct effect on the amplitude of the memory functions. In addition to this, as one moves dorsally in the pMN domain, the steady state concentration of Pax6 is increasing (Fig. 3.2E, also observed experimentally). This increase causes the system to become insensitive to fluctuations of Pax6 proteins as Pax6 binding sites are typically already saturated. This phenomenon is at work also in the other memory function amplitudes

that change across neural tube position as seen in Fig. 3.3A-D.

3.2.2. Nonlinear memory analysis

We next examined the second order memory functions. These provide corrections to the linear memory terms and encode information about how the bulk nodes drive the system towards a steady state. Quantitative accuracy in capturing the dynamics of the full nonlinear network will also be improved to the extent possible within an expansion around a steady state, but such accuracy is not our main focus – it is the additional insights gained from the memory functions that we are after.

It is apparent that once memory terms are included, different trajectories in the subnetwork plane *cross*. This is visually most obvious with the full nonlinear memory terms (Fig. 3.4C,G) and consistent with the full dynamics (Fig. 3.4D,H). The crossings arise from the fact that, in the presence of memory, systems in the same state can follow different trajectories towards a steady state depending on their past. In the full dynamics, the additional information as to which path a system will take is contained in the bulk species, which effectively hold the information on the past of the system. In the case of the memory functions, the memory is not contained in network nodes outside of the subnetwork, but is represented explicitly via the memory terms, with the time difference dependence of the memory functions indicating, for example, the timescale of the memory. This offers a useful abstraction: while it is not easy to visualize the four-dimensional concentration space for the full dynamics, the projection approach reduces the system to two components with memory of their own past.

In the nonlinear memory functions, we can assess the importance of the link from Olig2 to Pax6 by considering the squared deviations of Olig2: the Pax6 channel contribution to this memory necessarily involves signals being propagated via this Olig2$\rightarrow$Pax6 link. We find that the Pax6 channel makes a very small contribution to the nonlinear memory

amplitude in comparison to the Irx3 channel. This is shown in Fig. 3.3E for the p3 domain but is true also for all other states. We additionally checked the time-dependence of the memory function: from Fig. 3.3F one sees that the relative contribution of the Pax6 channel to the nonlinear memory of (past) Olig2 squared on Nkx2.2 remains small at all time-differences. Together with our results above for linear memory functions, we thus conclude that the repression of Pax6 by Olig2 is dispensable for maintaining an established dorsal-ventral pattern. We plot an example of nonlinear memory functions (Fig. 3.3G,H) to illustrate that their dependence on time difference can include non-monotonicities and sign changes that mean memory amplitudes do not tell the full story of the effect of nonlinear memory. The effects of including the memory terms can be confirmed and further visualised by considering the effective drift vector, which contains the rates of change of the concentrations of Nkx2.2 and Olig2. We describe this approach in 3.2.3, where we also discuss further how specific contributions of the nonlinear memory affect the dynamics, for example in the approach to the state in the p2 domain.

3.2.3. Further analysis of memory effects

In this subsection we give additional detail of the wide variety of effects that memory functions contribute to the behaviour of a system. Specifically, we explore the effect of the contributions from nonlinear memory terms and provide an alternative visualisation using a first order time expansion of the effective drift of the system.

Nonlinear memory effects

In Fig. 3.4 we show trajectory plots comparing the predictions without memory (left column) to those with linear and linear+quadratic memory (middle columns) and to the dynamics of the full network. Focussing directly onto the description with all memory terms included, we can gain insight into what drives the distinct dynamics of Fig. 3.4C,G by

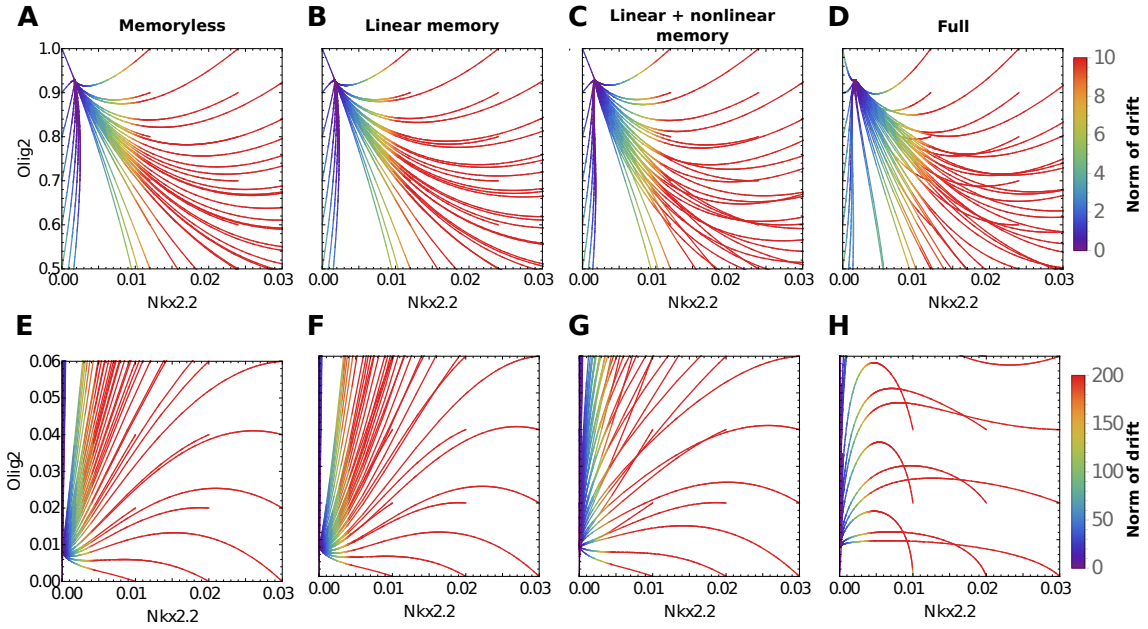


Figure 3.4: Trajectories approaching a steady state, with and without memory. Trajectories approaching a high Olig2 state in the pMN domain from a range of initial conditions, with no memory (**A**), with only linear memory (**B**), with linear and nonlinear memory (**C**), and the full dynamics (**D**). Trajectories approaching a low Olig2, low Nkx2.2 state in the p2 domain from different initial conditions, with no memory (**E**), with only linear memory (**F**), with linear and nonlinear memory (**G**), and the full dynamics (**H**). All figures are parametric plots, showing Nkx2.2 and Olig2 concentrations on the x- and y-axis, respectively, with time as curve parameter. The trajectories are coloured to represent the norm of the drift vector of the system as indicated by the colour scale; high values indicate the system is evolving quickly while the opposite is true for low values. Scalebars on the right apply to the corresponding row. Path crossing can be seen in (**C**–**D**) & (**G**–**H**), illustrating the importance of nonlinear memory terms to reproduce qualitative features of the full thermodynamic equations.

analysing the respective nonlinear memory functions, specifically their amplitude (value at $\Delta t = 0$) as shown in Fig. 3.5A-D. We note that in most cases the memory contributions arise via Irx3. (In the case of memory of Olig2 this is the only possibility in any case.) A clear exception to this is the case of nonlinear Nkx2.2 memory to itself (Fig. 3.5A) where the memory is dominated ventrally by the Pax6 channel and dorsally by the Irx3 channel. These memory functions also change with time difference, but it turns out that they decay on similar timescales so that the memory amplitude gives a reasonable characterization of their strength. We observe that Olig2 does not produce

memory effects that are substantial enough to change the dynamics (Fig. 3.5B,D): the dominant nonlinear memory effects come from Nkx2.2 (Fig. 3.5A,C). We see further that Nkx2.2 represses both itself (negative nonlinear memory amplitude, Fig. 3.5A) and, even more strongly, Olig2 (Fig. 3.5C). These two terms combined mean that at high levels of Nkx2.2, the memory functions will impede Olig2 from increasing as quickly as it would in cases without nonlinear memory, while simultaneously decreasing the concentration of Nkx2.2. Both effects are consistent with what we observe in Fig. 3.4C, D.

In the p2 domain the memoryless and linear memory predictions for trajectories are quite close, with levels of Nkx2.2 and Olig2 decreasing on similar timescales towards the steady state (Fig. 3.4E–H). Nonetheless, plots showing the norm of the effective drift indicate the trend that memory pushes the system more quickly towards low Nkx2.2 (Fig. 3.6D,E, see below). Including the nonlinear memory then further enhances this tendency: the levels of Nkx2.2 drop more rapidly than Olig2 on the approach to the steady state (Fig. 3.4G,H & Fig. 3.6F). The behaviour is not identical to but mimics that of the full thermodynamic model: the nonlinear memory effects contribute qualitative information about the behaviour of the system further away from the steady state. This points to memory terms being important for first reducing levels of Nkx2.2 before reducing Olig2 levels.

Understanding why this is the case requires a more detailed analysis of the memory effects. The memory amplitudes show that Nkx2.2 is not subject to substantial memory effects (Fig. 3.5E,F) and the largest memory effects act on Olig2 (Fig. 3.5G,H). At first glance, the latter memory amplitudes suggests that Nkx2.2 substantially represses Olig2 while Olig2 activates itself to a lesser degree. However, on closer inspection of the temporal dependence of the memory functions, the repression by Nkx2.2 is a short pulse that for larger Δt turns into an activation of Olig2 (Fig. 3.4F). Meanwhile, the Olig2 self-activation is a sustained signal that lasts much longer than the repression by Nkx2.2.

When taken together, these two memory terms (combined with cross terms, not shown) promote activation of Olig2 from high levels of Nkx2.2 even after a substantial time difference has elapsed, of the order of the decay time of the linear memory functions. This is consistent with what we observe in Fig. 3.4G,H: the initial high levels of Nkx2.2 lead to Olig2 levels being sustained via the memory, while Nkx2.2 decays in a way almost unaffected by memory. Eventually Olig2 also decays, once enough time has passed for the nonlinear memory effects to fade away.

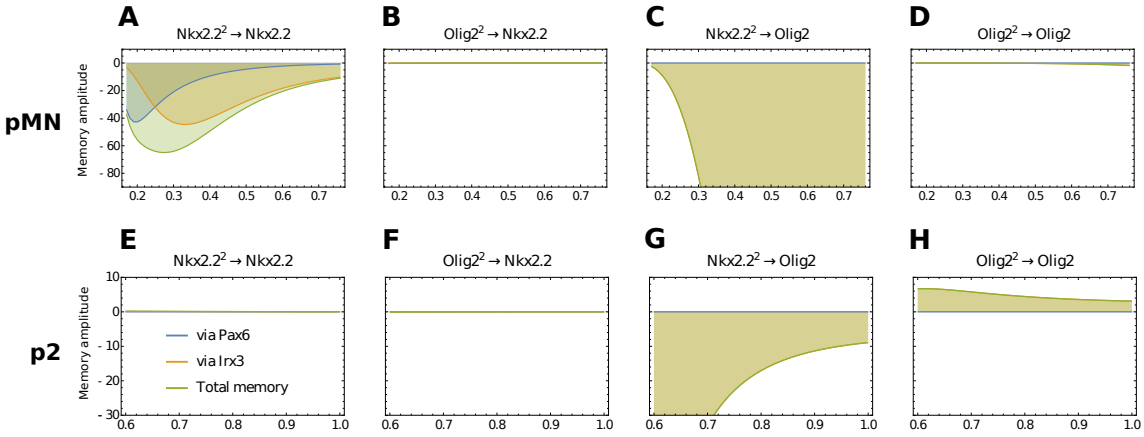


Figure 3.5: Nonlinear memory amplitudes. (**A**-**D**) Nonlinear memory amplitudes across neural tube positions within the pMN domain. The plot titles indicate the type of memory, *e.g.* (**C**) shows memory of (past) Nkx2.2 squared fluctuation on Olig2. Memory effects to past Olig2-fluctuations in the pMN domain are clearly negligible. (**E**-**H**) Nonlinear memory amplitudes across neural tube positions within the p2 domain. Nkx2.2 visibly receives very little quadratic memory influence in this domain. The x-axis represents neural tube position in all plots. Blue and yellow lines indicate the decomposition into Pax6 and Irx3 channels, while green lines indicate the total memory.

Calculation of effective drift in the presence of memory

It is useful to be able to visualize the effects of memory terms in the projected equations in terms of an effective drift. This is possible at least perturbatively for short times, as we now show. We illustrate the method for the linearized dynamics, where the projected

equation (2.3) without the random force, read in vector form:

$$\partial_t \boldsymbol{x}^{\mathrm{T}} = \boldsymbol{x}(t)^{\mathrm{T}} \boldsymbol{\Omega} + \int_0^t dt' \boldsymbol{x}(t)^{\mathrm{T}} \boldsymbol{M}(t-t') \tag{3.2}$$

For small times t, the memory function can be treated as approximately constant so that one obtains in an expansion to first order in t:

$$\int_0^t dt' \boldsymbol{x}(t)^{\mathrm{T}} \boldsymbol{M}(t-t') \simeq t\, \boldsymbol{x}(t)^{\mathrm{T}} \boldsymbol{M}(0) \tag{3.3}$$

Combining with the rate matrix term gives the effective drift

$$\partial_t \boldsymbol{x}^{\mathrm{T}} = \boldsymbol{x}(t)^{\mathrm{T}} [\boldsymbol{\Omega} + t\boldsymbol{M}(0)] \tag{3.4}$$

On the right hand side one has a function of the current concentrations only, as previously stated. Given that this is a linear expansion in t we refrain from exploring the system too far away from $t = 0$. Higher order expansions in t could be performed yet the first order is enough to demonstrate the qualitative contributions of the memory terms.

We illustrate the approach in Fig. 3.6A-F with contour plots of the norm of the effective drift vector on the r.h.s. of (3.4), for time $t = 0.8$. Bearing in mind that the system will pass quickly through regions where the drift is high and spend most of its time in regions where it is low, these plots confirm the effects seen in the trajectory plots: inclusion of the memory terms causes the low-drift region to shift to lower Nkx2.2 concentrations, *i.e.* the system will more rapidly reduce Nkx2.2 and then spend more time increasing Olig2 at small Nkx2.2.

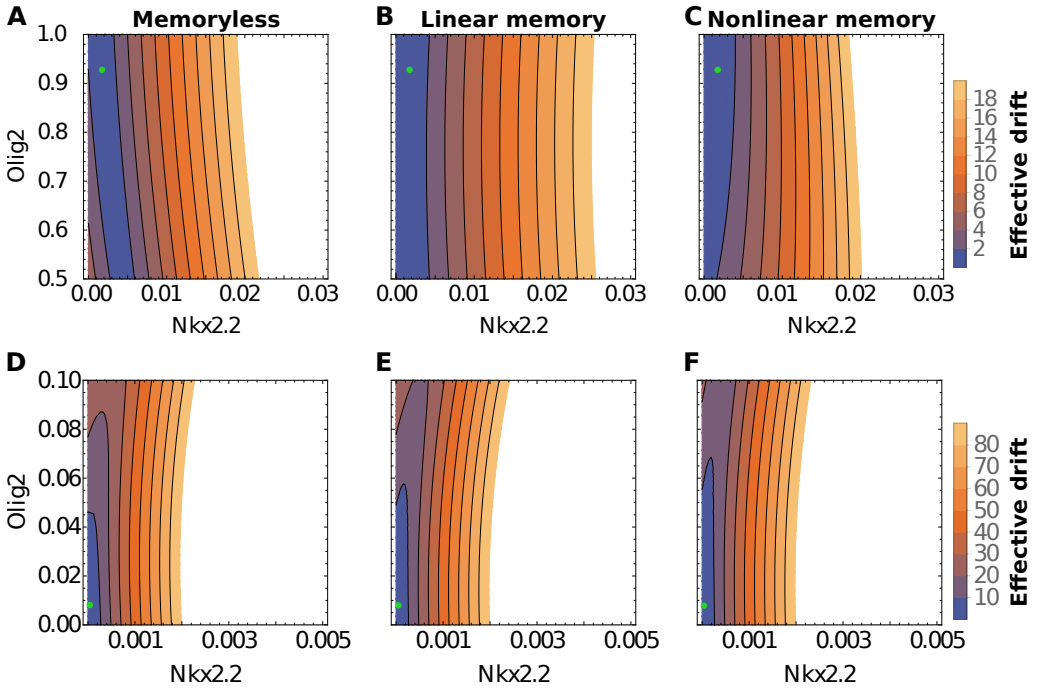


Figure 3.6: Colour map and contour plots of the norm of the effective drift, in conditions identical to those in Fig. 3.4A-C for the first row (pMN domain) and Fig. 3.4E-G (p2 domain) for the second row. (**A**,**D**) Memoryless drift, (**B**,**E**) drift with linear memory, (**C**,**F**) drift with nonlinear memory. The scalebar on the right applies to each entire row. Effective drift norms have been calculated for time $t = 0.8$.

3.3. Exploration of network properties

The analysis of both linear and nonlinear memory functions in the previous section suggested that the repression of Pax6 by Olig2 is not critical for the dynamics of the steady states observed in neural tube patterning. We based this on the observation that the amplitude of memory that is transmitted through this link is relatively small compared to the other memory functions, for all biologically relevant steady states.

To test this prediction of the memory function analysis, we removed the repressive link from Olig2 to Pax6 in the full model. Consistent with our prediction of the relative insignificance of the link from Olig2 to Pax6, the resulting steady state concentrations of the simulations are similar to those in the original system (Fig. 3.7A). The only

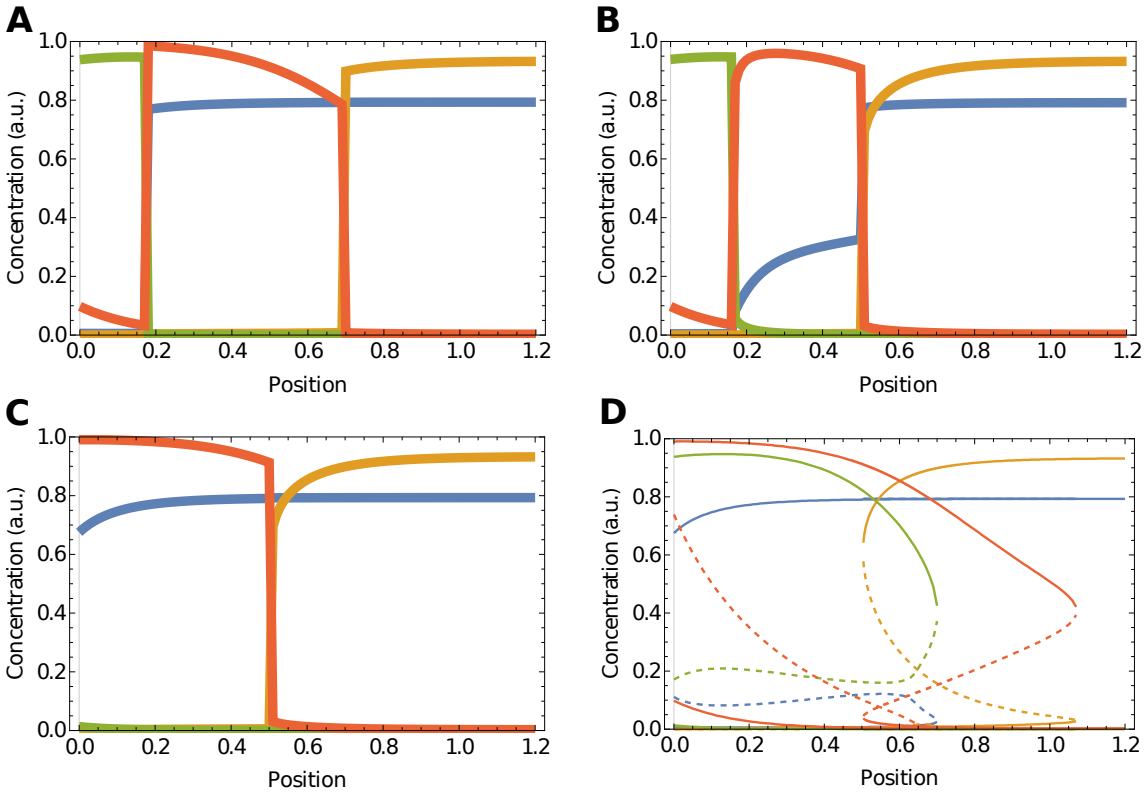


Figure 3.7: Olig2 repression of Pax6 increases robustness to initial conditions. (**A**) Patterning without the repressive link from Olig2 to Pax6, showing steady states reached from standard initial conditions. Compared to the full network, the qualitative domain structure is conserved. (**B**) Patterning for the full network for initial conditions with high levels of Pax6 and Irx3 is qualitatively identical to low initial levels of Pax6 and Irx3. Initial conditions $[\text{Pax6}] = [\text{Irx3}] = 1$. (**C**) Patterning without the repressive link from Olig2 to Pax6 for initial conditions with high levels of Pax6 and Irx3. The p3 domain is lost as a consequence of the different initial conditions. (**D**) Bifurcation diagram of the network without Olig2-Pax6 repression. The region in which the pMN state (Olig2 high) is stable has expanded ventrally, thus making the ventralmost region bistable.

qualitative change is the increased level of Pax6 between neural tube position 0.2 and 0.7, which results from the loss of the repressive link. But the spatial aspects, such as the positional sequence of genes and qualitative concentrations, of the domains are otherwise unaltered. Similar effects have been observed in the case of entire genes in Verd et al. (2019), termed "node insensitivity".

Since the repressive link between Olig2 and Pax6 has been experimentally documented (Balaskas et al., 2012, Kutejova et al., 2016, Zhou and Anderson, 2002) but from our

analysis is not required for the dynamics around the steady state, we sought to understand what purpose it might serve during development. We performed simulations from varying initial conditions for both WT (Fig. 3.7B) and the system lacking the link (Fig. 3.7C). These show that the p3 domain is reached from only a small range of initial conditions following the removal of the repressive link between Olig2 and Pax6. To rationalise this, we performed a bifurcation analysis (Fig. 3.7D). This indicated that the removal of the Olig2 inhibition of Pax6 markedly increased the range of positions at which the Olig2 steady state is present, thus facilitating its invasion into the Nkx2.2 domain. In particular, under initial conditions of high Pax6 and Irx3, the pMN domain is induced ventrally. In the unperturbed network it is not possible to reach this steady state in the ventralmost part because the system is monostable, allowing only the p3 domain to arise. On the other hand, in the absence of Olig2 inhibition of Pax6, the ventral region is bistable. Hence, depending on the initial conditions a pMN fate can be reached (Fig. 3.7D). Taken together, this analysis suggests that the repressive link from Olig2 to Pax6 contributes to the robustness of the system for patterning the neural tube.

A potential explanation for the seemingly dispensable regulatory interaction between Olig2 and Pax6 arises from considering the normal development of the neural tube. During embryonic neural tube development, neural progenitors are generated by the process of neural induction and this initiates the expression of neural genes including Pax6 and Irx3. At the same time, cells of the notochord (which underlies the ventral midline of the neural tube) begin to secrete Shh. Shh spreads into the neural tube and neural progenitors respond to the signal. The consequence is that in the ventral neural tube neural progenitors begin responding to Shh at intermediate levels of Pax6 and Irx3. A more appropriate initial condition, representative of the in vivo situation, is for Pax6 and Irx3 to be somewhere between 0 and maximum. Moreover, heterogeneity between cells, as well as variation in the exact timing of developmental events along the rostral-

caudal axis of the embryo, means that the precise initial conditions – the levels of Pax6 and Irx3 when cells respond to Shh signalling – will vary. Hence the robustness provided by the Olig2-Pax6 link may play an important role in ensuring reliable pattern formation in the developing neural tube.

3.4. Discussion

We have developed a generally applicable method for applying Zwanzig-Mori projections to transcriptional networks that allows the analysis of subnetworks extracted from a larger network. We demonstrated the approach on a simple genetic cross repressive motif, to illustrate the minimal example of a memory function, and on a more complicated, but well-characterised, transcriptional network operating in the ventral part of the vertebrate neural tube. This showed how the method allows the function and importance of specific links within a network to be defined in an intuitive manner. We used these insights to identify structural features of the neural tube network that appear primarily to increase the robustness of the network to initial conditions, rather than maintain its steady state.

The projection technique is straightforward to implement using the methodology described, and can be applied to any GRN following the thermodynamic formalism (including activation, repression, competitive and cooperative binding). Further protein-protein interactions could readily be incorporated as these would be represented by simple first order (unary) or second order (binary) interactions. Thus, in addition to gene regulation, nonlinear protein-protein interactions mediated by enzymes such as those found in signal transduction pathways could be included in the analysis (Rubin and Sollich, 2016). It would also be possible to include Hill functions as long as these possess integer exponents (Thomas et al., 2012b). Note that our treatment relies on mass action kinetics, which implicitly assumes that the different molecular species are well mixed by diffusion processes that are faster than any of the reaction kinetics (Smith and Grima, 2016). The

size of the matrices involved in constructing the memory functions scales as the square of the number of bulk species, which in our experience means that networks with up to 200 nodes can be investigated without difficulty. Thus, the generality of the method and the ability to implement it algorithmically provide a comprehensive mathematical toolkit to simplify and analyse dynamical systems describing a range of cellular and molecular processes.

An example of the type of insight provided by the projection method came from our analysis of the neural tube network. We focused on a subnetwork comprising Nkx2.2 and Olig2. Decomposition of the memory functions suggested that an experimentally documented repression of Pax6 by Olig2 (Zhou and Anderson, 2002) plays only a minor role in sustaining the steady state pattern. Consistent with this prediction, simulations in which this regulatory link had been eliminated in the full model showed qualitatively unchanged steady states. We note that removal of the repressive link between Olig2 and Pax6 is distinct from the removal of Olig2 itself (Balaskas et al., 2012, Cohen et al., 2013, 2014). In the Olig2 mutant, the repressive effect of Olig2 on both Nkx2.2 and Pax6 are eliminated whereas the removal of the Pax6-Olig2 link leaves the repressive effect of Olig2 on Nkx2.2 intact. The memory function analysis thus raised the question of what purpose the Olig2-Pax6 regulation might serve and prompted us to explore the transient system dynamics, specifically the effect of the initial conditions, which are unknown in the *in vivo* system. Simulations in the absence of the Olig2-Pax6 repression, compared to the full model, revealed a marked increase in sensitivity to these initial conditions. Comparison of the bifurcation diagrams of the systems, with and without the Pax6-Olig2 link, indicated that the well-defined Nkx2.2 monostable region (p3 domain) – which appears at high levels of Shh signalling in the full system – was replaced by a region of bistability for Nkx2.2 and Olig2 in the absence of the Olig2-Pax6 regulation. In this latter case the steady state induced by high levels of signal therefore depends

on the initial concentrations of Pax6 and Irx3. This suggests that, while not necessary to maintain steady states, the Olig2-Pax6 regulation ensures access to the appropriate steady states irrespective of the initial levels of Pax6 and Irx3. As these may differ between cells and at different locations along the rostral-caudal axis of the neural tube, the Olig2-Pax6 regulatory link might make the system less sensitive to these variations. This implies that the primary purpose of Olig2 repression of Pax6 is to increase the robustness of pattern formation.

Alternative implementations of the Zwanzig-Mori projection (Chorin et al., 2000, Nicolis, 1980, Zwanzig, 2001) have been developed that do not rely on expanding around a fixed steady state. While these alternative methods may provide a solution to the specific limitations of the last two chapters' Zwanzig-Mori projection method, the nonlinear memory functions they produce are much richer, see (Chapter 4). Further work will be needed to fully understand the more complex information they encode. Projection methods might also provide a way to analyse systems for which only partial information is available. Given a known subnetwork that is not able to fully reproduce experimental observations, the incorporation of memory functions provides a means to explore the possible effects of unknown factors. This could then be used to identify plausible network structures that generate these memory functions. Taken together therefore, the projection approach provides tools to simplify, visualise and explore the behaviour of large networks that would otherwise be difficult to analyse in their entirety.

4

A projection approach far from steady state

4.1. Motivation

In dynamical systems such as protein interaction networks and gene regulation networks, the increase in complexity with system size can quickly lead to difficulties. These can be of a numerical nature, preventing efficient solutions. More importantly large dynamical systems also pose conceptual challenges, with a qualitative understanding of their function, e.g. in response to external signals, being hard to achieve. For this reason

many different approaches to reduce the dimensionality of a system have been developed. Some approaches to this are by deriving a new form from the original equations or by observing which reduced systems fit certain desired dynamics or targets. The task of dimensionality reduction becomes substantially more difficuly when dealing with emergent behaviour such as multistability or oscillations. As highlighted in the previous chapter, the Zwanzig-Mori formalism provides an exact dimensionality reduction based on a choice of "subnetwork" observables, with the remaining variables then taken as the environment or "bulk" (Mori, 1965, Nakajima, 1958, Zwanzig, 1961). This approach, specifically its nonlinear version (Zwanzig, 2001), was originally developed for the dynamics of physical systems but later generalised by Chorin and coworkers (Chorin et al., 2000, 2002). Their approach allows one to formally coarse grain an arbitrary dynamical system to a smaller set of observables. As we have seen in Chapters 2 & 3, the elimination of the bulk degrees of freedom leads to memory effects, where the subnetwork state at some time affects the bulk which eventually feeds back to the subnetwork at a later time. The memory functions that govern these effects are, however, generally impossible to calculate in closed form (Chorin et al., 2000, 2002). While approximate expressions can be derived in special cases (Beck et al., 2009, Chorin and Stinis, 2006, Gouasmi et al., 2017, Stinis, 2006, Weinan et al., 2008), this restricts the applicability of the formalism. Progress can be made by, for example, expanding the dynamical equations around a fixed point (Bravi and Sollich, 2017b, Rubin et al., 2014), but for multistable systems the memory functions obtained in this way can then not be used effectively.

In this chapter we develop a method that allows the calculation of memory functions for generic multistable dynamical systems. We make only one assumption: the bulk must not generate fixed points beyond those of the subnetwork, more specifically it must have a unique steady state for any subnetwork state such as in (Gouasmi et al., 2017). This is a natural condition: the subnetwork must be large enough to produce all

fixed points itself, otherwise coarse graining cannot succeed. We take as the baseline for the coarse grained or "projected" dynamics the quasi-steady state approximation, where the bulk is always in steady state with the current subnetwork state. This would be exact in the limit where the dynamics in the bulk is infinitely fast. Memory effects improve the projected subnetwork description when the bulk is not in fact fast. Our main technical result (Sec. 4.2) is an explicit solution for the memory functions capturing these memory effects, derived in a systematic expansion around the Quasi-steady state (QSS) approximation that can be viewed as a similar approach to (Gouasmi et al., 2017), where we go on to solve the system without need of prior knowledge regarding its dynamics. The method is broadly applicable to dynamical systems so long as condition of unique bulk steady states is met.

We then apply our method (Sec 4.3) to show its efficiency and precision, for a range of model dynamical systems presenting different types of complex emergent behaviour as well as a well parametrised biologically relevant network. Finally (Sec. 4.4) we show that further to reducing dimensionality our method can detect aspects of a system's behaviour that would otherwise be difficult to access.

4.2. Mathematical derivation

4.2.1. Baseline definitions

One starts from a system with degrees of freedom $\boldsymbol{x}$ evolving deterministically in time according to some nonlinear functions $\boldsymbol{R}$:

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{R}(\boldsymbol{x}) \tag{4.1}$$

We define the "flow" (vector) $\boldsymbol{\phi}(\boldsymbol{x}, t)$ as the state the system reaches at time t if it starts in some initial state $\boldsymbol{x}$; this function thus obeys $\boldsymbol{\phi}(\boldsymbol{x}, 0) = \boldsymbol{x}$ and $\frac{\partial}{\partial t}\boldsymbol{\phi}(\boldsymbol{x}, t) =$

$\boldsymbol{R}(\boldsymbol{\phi}(\boldsymbol{x},t))$. We want to understand the dynamics of some chosen set of observables that we denote by the vector $\boldsymbol{A}$. Such observables are functions of the state of the system, which we write as $\boldsymbol{A}(\boldsymbol{x})$. By analogy with the definition of $\boldsymbol{\phi}$, the time-dependent observables are then taken as

$$\boldsymbol{A}(\boldsymbol{x},t) = \boldsymbol{A}(\boldsymbol{\phi}(\boldsymbol{x},t)) \tag{4.2}$$

so that $\boldsymbol{A}(\boldsymbol{x},t)$ gives the value of the observables at time t if the system was initially in state $\boldsymbol{x}$. The resulting time evolution of the observables can again be described by a differential equation:

$$\frac{\partial}{\partial t}\boldsymbol{A} = \sum_i \frac{\partial \phi_i}{\partial t}(\boldsymbol{x},t)\frac{\partial \boldsymbol{A}}{\partial \phi_i}(\boldsymbol{\phi}(\boldsymbol{x},t)) \tag{4.3}$$

$$= \sum_{ij} R_j(\boldsymbol{x})\frac{\partial \phi_i}{\partial x_j}(\boldsymbol{x},t)\frac{\partial \boldsymbol{A}}{\partial \phi_i}(\boldsymbol{\phi}(\boldsymbol{x},t)) \tag{4.4}$$

$$= \sum_j R_j(\boldsymbol{x})\frac{\partial}{\partial x_j}\boldsymbol{A}(\boldsymbol{\phi}(\boldsymbol{x},t)) = L\boldsymbol{A}(\boldsymbol{x}) \tag{4.5}$$

with the Liouvillian

$$L = \sum_i R_i(\boldsymbol{x})\frac{\partial}{\partial x_i} \tag{4.6}$$

which is a linear differential opperator. In going from (4.3) to (4.4) we have used the fact that $\partial\boldsymbol{\phi}/\partial t = \boldsymbol{R}(\boldsymbol{\phi})$ can also be written as $\sum_j R_j(\boldsymbol{x})\partial\boldsymbol{\phi}/\partial x_j$. This follows from considering a small time interval dt as being added to the beginning rather than the end of a trajectory: $\boldsymbol{\phi}(\boldsymbol{x}, t+dt) \approx \boldsymbol{\phi}(\boldsymbol{x} + \boldsymbol{R}(\boldsymbol{x})dt, t)$.

The general setup above of course requires one to track the full $\boldsymbol{x}$-dependence of the chosen observables $\boldsymbol{A}(\boldsymbol{x},t)$. To achieve a reduction in dimensionality, Chorin (Chorin et al., 2000) assumes that the $\boldsymbol{x}$ are determined by the $\boldsymbol{A}$ at least statistically, i.e. have

some probability distribution that (only) depends on the current value of $\boldsymbol{A}$. Averages (expectations) over this distribution are written as $E[\cdot|\boldsymbol{A}]$, and the average evolution of $\boldsymbol{A}$ is governed by $\boldsymbol{v}(\boldsymbol{A}) = E[L\boldsymbol{A}(\cdot)|\boldsymbol{A}]$. Chorin (Chorin et al., 2000, 2002) showed that the corrections to this in the actual time evolution take the form of a memory term and a so-called random force $\boldsymbol{r}$, giving the general form for the time evolution of $\boldsymbol{A}$ as:

$$\frac{d}{dt}\boldsymbol{A} = \boldsymbol{v}(\boldsymbol{A}) + \int_0^t dt' \; \boldsymbol{M}(\boldsymbol{A}(t'), t - t') + \boldsymbol{r} \tag{4.7}$$

The memory function $\boldsymbol{M}(\boldsymbol{A}(t'), t - t')$ depends on time difference $\tau = t - t'$ and – nonlinearly – on the past observable value $\boldsymbol{A}(t')$. Its evolution with τ is governed by the *deviations* of the drift from $\boldsymbol{v}(\boldsymbol{A})$; this evolution reads for a general observable $g(\boldsymbol{x}, \tau)$

$$\frac{\partial}{\partial\tau} g(\boldsymbol{x}, \tau) = Lg(\boldsymbol{x}, \tau) - E[(Lg)(\cdot, \tau)|\boldsymbol{A}(\boldsymbol{x})] \tag{4.8}$$

The memory function is obtained in particular from the observable that measures exactly such fluctuations in the drift of the observables $\boldsymbol{A}$, this (vector) observable is defined as:

$$\boldsymbol{F}(\boldsymbol{x}) = (L\boldsymbol{A})(\boldsymbol{x}) - E[L\boldsymbol{A}(\cdot)|\boldsymbol{A}(\boldsymbol{x})] \tag{4.9}$$

From $\boldsymbol{F}(\boldsymbol{x})$ one then defines an $\boldsymbol{F}(\boldsymbol{x}, \tau)$ which evolves according to (4.8) from the initial condition $\boldsymbol{F}(\boldsymbol{x}, 0) = \boldsymbol{F}(\boldsymbol{x})$, thus the memory function is given explicitly as $\boldsymbol{M}(\boldsymbol{A}, \tau) = E[L\boldsymbol{F}(\cdot, \tau)|\boldsymbol{A}]$. The random force itself is $\boldsymbol{r}(\boldsymbol{x}, t) = \boldsymbol{F}(\boldsymbol{x}.t)$ and has a vanishing average at all times, $E[\boldsymbol{r}(\cdot, t)|\boldsymbol{A}] = 0$ (Chorin et al., 2002) we will not account for it in the following, as the random force vanished when the bulk starts in QSS. While in that study, steady state dynamics are discussed, this is not required for the formalism to be defined.

4.2.2. Subnetwork dynamics

With the random force discarded as above, Eq. (4.7) is a closed equation for the time evolution of the observables $\boldsymbol{A}$ and so achieves the desired dimensionality reduction. However, the memory function cannot in general be calculated in any closed form. We now show that this *can* be done, within a systematic approximation, for subnetwork dynamics. By this we mean that we consider as the observables $\boldsymbol{A} = \boldsymbol{x}^{\mathrm{s}}$ a subset of $\boldsymbol{x}$, e.g. the concentrations of molecular species in a subnetwork of a large gene regulatory network. We denote the degrees of freedom in the rest of the network, the bulk, by $\boldsymbol{x}^{\mathrm{b}}$ and write the components of the general time evolution (4.1) explicitly as

$$\frac{d\boldsymbol{x}^{\mathrm{s}}}{dt} = \boldsymbol{R}^{\mathrm{s}}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) \tag{4.10}$$

$$\frac{d\boldsymbol{x}^{\mathrm{b}}}{dt} = \boldsymbol{R}^{\mathrm{b}}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) \tag{4.11}$$

The Liouvillian then splits accordingly into

$$L = \sum_s R_s(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) \frac{\partial}{\partial x_s} + \sum_b R_b(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) \frac{\partial}{\partial x_b} \tag{4.12}$$

with the sums running over subnetwork and bulk species, respectively. Here and below subscripts always indicate individual species, while vectors with 's' and 'b' superscripts collect all subnetwork and bulk quantities, respectively. The generic observable time evolution (4.3) $(\partial/\partial t)\boldsymbol{x}^{\mathrm{s}} = L\boldsymbol{x}^{\mathrm{s}}$ then just reduces to (4.10) as it should. To summarise, we can write the time evolution of the subnetwork species as:

$$\frac{\partial}{\partial t}\boldsymbol{A} = \boldsymbol{R}^{\mathrm{s}}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) = L\boldsymbol{A} \tag{4.13}$$

We now need to choose how to define the expectation $E[\cdot|\boldsymbol{x}^{\mathrm{s}}]$. We do this so that without the memory kernel, the reduced equation (4.7) corresponds to the simplification

where the bulk dynamics equilibrates rapidly to any prevailing subnetwork state $\boldsymbol{x}^{\mathrm{s}}$, reaching a quasi steady state (QSS) value $\boldsymbol{x}^{\mathrm{b}*}$ defined by $d\boldsymbol{x}^{\mathrm{b}}/dt = 0$ or

$$\boldsymbol{R}^{\mathrm{b}}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*}(\boldsymbol{x}^{\mathrm{s}})) = 0 \tag{4.14}$$

As motivated in the introduction we will assume that this condition determines a unique bulk QSS $\boldsymbol{x}^{\mathrm{b}*}(\boldsymbol{x}^{\mathrm{s}})$ for any $\boldsymbol{x}^{\mathrm{s}}$, in practice we only check that such condition is true in the explored phase space. Our assumption therefore does not affect the existence or properties of the fixed points of the original system. The expectation required to construct the reduced equation (4.7) is taken accordingly as

$$E[g(\cdot)|\boldsymbol{x}^{\mathrm{s}}] = g(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*}(\boldsymbol{x}^{\mathrm{s}})) \tag{4.15}$$

i.e. by taking $\boldsymbol{x}^{\mathrm{s}}$ as prescribed and inserting for $\boldsymbol{x}^{\mathrm{b}}$ its QSS value. The average drift $\boldsymbol{v}(\boldsymbol{A}) = E[L\boldsymbol{A}(\cdot)|\boldsymbol{A}] = E[\boldsymbol{R}^{\mathrm{s}}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}})|\boldsymbol{x}^{\mathrm{s}}]$ now evaluates directly from (4.15) as

$$\boldsymbol{v}(\boldsymbol{x}^{\mathrm{s}}) = \boldsymbol{R}^{\mathrm{s}}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*}(\boldsymbol{x}^{\mathrm{s}})) \tag{4.16}$$

This is the QSS or 'fast bulk' approximation to the subnetwork dynamics. Our main interest in the following lies in understanding the memory effects that account for the fact that the bulk will not in general be fast but evolve on a timescale comparable to that of the subnetwork. To determine the memory function, we start from the definition of $\boldsymbol{F}(\boldsymbol{x})$, which from (4.9) has components

$$F_s(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) = R_s(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) - E[R_s(\cdot)|\boldsymbol{x}^{\mathrm{s}}] \tag{4.17}$$

$$= R_s(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) - R_s(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*}(\boldsymbol{x}^{\mathrm{s}})) \tag{4.18}$$

The main challenge is now to calculate the evolution of this observable in time according

to (4.8). This is not feasible in general but we can develop a systematic approximation by *linearising* in deviations of the bulk degrees of freedom from the QSS, which we write as

$$F_s(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}, t) \approx \sum_b (x_b - x_b^*(\boldsymbol{x}^{\mathrm{s}})) f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) \tag{4.19}$$

The problem then reduces to finding the evolution of $f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau)$ from the initial condition $f_{bs}(\boldsymbol{x}^{\mathrm{s}}, 0) \equiv f_{bs}^0(\boldsymbol{x}^{\mathrm{s}})$ obtained by linearising (4.18):

$$f_{bs}^0(\boldsymbol{x}^{\mathrm{s}}) = \frac{\partial R_s}{\partial x_b} \tag{4.20}$$

where the derivatives here and below are evaluated at $(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*}(\boldsymbol{x}^{\mathrm{s}}))$.

4.2.3. Memory evolution over time

If we make use of our expansion (4.19) we find for the first term on the RHS of (4.8)

$$\begin{aligned} LF_s = &\sum_b R_b(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) \\ &+ \sum_{s'} R_{s'}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) \sum_b (x_b - x_b^*) \frac{\partial}{\partial x_{s'}} f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) \\ &- \sum_{s'} R_{s'}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) \sum_b \frac{\partial x_b^*}{\partial x_{s'}} f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) \end{aligned} \tag{4.21}$$

where the two last lines arise by differentiating the product $(\partial/\partial x_{s'})(x_b - x_b^*) f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau)$ and we have dropped the $\boldsymbol{x}^{\mathrm{s}}$-dependence of $\boldsymbol{x}^{\mathrm{b}*}$ for brevity. The second term on the RHS of (4.22) is the expectation of this, obtained by replacing $\boldsymbol{x}^{\mathrm{b}}$ by $\boldsymbol{x}^{\mathrm{b}*}$:

$$\begin{aligned} E[LF_s|\boldsymbol{x}^{\mathrm{s}}] = &\sum_b R_b(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*}) f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) \\ &- \sum_{s'} R_{s'}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*}) \sum_b \frac{\partial x_b^*}{\partial x_{s'}} f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) \end{aligned} \tag{4.22}$$

Putting the two together gives for the time evolution (4.8) of F_s

$$\begin{aligned}\frac{\partial F_s}{\partial \tau} =& \sum_b \left[R_b(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) - R_b(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*})\right] f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) \\ &+ \sum_{s'} R_{s'}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*}) \sum_b (x_b - x_b^*) \frac{\partial}{\partial x_{s'}} f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) \\ &- \sum_{s'b} \left[R_{s'}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) - R_{s'}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*})\right] \frac{\partial x_b^*}{\partial x_{s'}} f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau)\end{aligned} \tag{4.23}$$

For consistency with (4.19) we now linearise the square brackets in the first and third line of (4.23) in the following way:

$$\left[R_b(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) - R_b(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*})\right] \simeq \sum_{b'} \frac{\partial R_b}{\partial x_{b'}}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*})(x_{b'} - x_{b'}^*) \tag{4.24}$$

$$\left[R_{s'}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}}) - R_{s'}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*})\right] \simeq \sum_{b'} \frac{\partial R_{s'}}{\partial x_{b'}}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*})(x_{b'} - x_{b'}^*) \tag{4.25}$$

This leads us to a simplified version of (4.23):

$$\begin{aligned}\frac{\partial F_s}{\partial \tau} =& \sum_{bb'} \frac{\partial R_b}{\partial x_{b'}}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*})(x_{b'} - x_{b'}^*) f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) \\ &+ \sum_{s'b} R_{s'}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*})(x_b - x_b^*) \frac{\partial}{\partial x_{s'}} f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) \\ &- \sum_{s'b'b} \frac{\partial R_{s'}}{\partial x_{b'}}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*})(x_{b'} - x_{b'}^*) \frac{\partial x_b^*}{\partial x_{s'}} f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau)\end{aligned} \tag{4.26}$$

Comparing then with the time derivative of the original linearised formula (4.19) gives after appropriate relabelling of indices. We then replace $\boldsymbol{F}_s$ with the form $f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau)$, note that $f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau)$ includes already a b term, so we will relabel b to b' and viceversa for consistency, such that we can write it in a form that is comparable with (4.19).

Therefore we obtain the time derivative for f:

$$\begin{aligned}\frac{\partial}{\partial\tau}f_{bs}(\boldsymbol{x}^{\mathrm{s}},\tau) = &\sum_{b'}\frac{\partial R_{b'}}{\partial x_b}f_{b's}(\boldsymbol{x}^{\mathrm{s}},\tau)\\ &+\sum_{s'}R_{s'}\frac{\partial}{\partial x_{s'}}f_{bs}(\boldsymbol{x}^{\mathrm{s}},\tau)\\ &-\sum_{b's'}\frac{\partial R_{s'}}{\partial x_b}\frac{\partial x^*_{b'}}{\partial x_{s'}}f_{b's}(\boldsymbol{x}^{\mathrm{s}},\tau)\end{aligned} \tag{4.27}$$

4.2.4. Obtaining more compact general form

To obtain a simplified form, one defines $l_{bb'}$:

$$\frac{\partial}{\partial\tau}f_{bs}(\boldsymbol{x}^{\mathrm{s}},\tau) = \sum_{b'}l_{bb'}f_{b's}(\boldsymbol{x}^{\mathrm{s}},\tau) + \sum_{s'}R_{s'}\frac{\partial}{\partial x_{s'}}f_{bs} \tag{4.28}$$

We can use the fact that because bulk species are fast and are "slaves" to slow species, (4.14) holds for all $\boldsymbol{x}^{\mathrm{s}}$ so we can also differentiate with respect to $\boldsymbol{x}^{\mathrm{s}}$:

$$\frac{\partial R_{b''}}{\partial x_{s'}} + \sum_{b'}\frac{\partial R_{b''}}{\partial x_{b'}}\frac{\partial x^*_{b'}}{\partial x_{s'}} = 0 \tag{4.29}$$

We then simply define the Jacobian of $R_{b''}$:

$$J_{b''b'} = \frac{\partial R_{b''}}{\partial x_{b'}} \tag{4.30}$$

Thus, we write the required expression as:

$$\frac{\partial x^*_{b'}}{\partial x_{s'}} = -\sum_{b''}(\boldsymbol{J}^{-1})_{b'b''}\frac{\partial R_{b''}}{\partial x_{s'}} \tag{4.31}$$

Therefore one can extract $\boldsymbol{l}$ of the form:

$$l_{bb'} = J_{b'b} + \sum_{s'b''} (\boldsymbol{J}^{-1})_{b'b''} \frac{\partial R_{b''}}{\partial x_{s'}} \frac{\partial R_{s'}}{\partial x_b} \tag{4.32}$$

Where we have removed the unknown derivative of $\boldsymbol{x}^{\mathrm{b}*}$ in our solution.

4.2.5. Finding a solution for f

The next step is to find a solution $f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau)$ for the partial differential equation (4.28). This can be done using the method of characteristics as the equation is linear in $f_{bs}(\boldsymbol{x}^{\mathrm{s}},)$ and only involves first derivatives, and gives the closed form solution (see Appendix 4.A)

$$f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) = \sum_{b'} E_{bb'}(0, \tau) f^0_{b's}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}, \tau)) \tag{4.33}$$

Here the $E_{bb'}$ are elements of the time-ordered matrix exponential $\boldsymbol{E}(0,\tau) = e^{\int_0^\tau d\tau' \, \boldsymbol{l}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}, \tau'))}$ defined in (4.49) and the propagation in time is done with the flow ϕ_v corresponding only to the QSS drift $\boldsymbol{v}(\boldsymbol{x}^{\mathrm{s}})$. The algebraic process to prove that this form satisfies (4.28) is described in detail in Appendix 4.A.

4.2.6. Memory function

We can now finally determine the memory function on subnetwork species s, which from the general framework set out above is

$$M_s(\boldsymbol{x}^{\mathrm{s}}, \tau) = E[LF_s(\cdot, \tau) | \boldsymbol{x}^{\mathrm{s}}] \tag{4.34}$$

We already worked out the expectation in (4.22).

$$E[LF_s(\cdot, \tau) | \boldsymbol{x}^{\mathrm{s}}] \simeq E[L \sum_{b'} (x_{b'} - x^*_{b'}) f_{b's}(\boldsymbol{x}^{\mathrm{s}}, \tau) | \boldsymbol{x}^{\mathrm{s}}] \tag{4.35}$$

The first term, which we had kept previously to make the ensuing linearisation easier to see, actually vanishes because of (4.14), yielding

$$M_s(\boldsymbol{x}^{\mathrm{s}},\tau) = -\sum_{s'b'} R_{s'} \frac{\partial x^*_{b'}}{\partial x_{s'}} f_{b's}(\boldsymbol{x}^{\mathrm{s}},\tau) \tag{4.36}$$

Using again the identity (4.31) then produces our main result, a simple expression for the memory function:

$$M_s(\boldsymbol{x}^{\mathrm{s}},\tau) = \sum_{b'} c_{b'}(\boldsymbol{x}^{\mathrm{s}}) f_{b's}(\boldsymbol{x}^{\mathrm{s}},\tau) \tag{4.37}$$

where we have abbreviated

$$c_{b'}(\boldsymbol{x}^{\mathrm{s}}) = \sum_{s'} \sum_{b''} (\boldsymbol{J}^{-1})_{b'b''} \frac{\partial R_{b''}}{\partial x_{s'}} R_{s'} \tag{4.38}$$

These functions can be thought of as prefactors to the memory term. The general projected time evolution equation now takes the form

$$\frac{d}{dt} x_s = v_s(\boldsymbol{x}^{\mathrm{s}}(t)) + \int_0^t dt' \; M_s(\boldsymbol{x}^{\mathrm{s}}(t'), t-t') \tag{4.39}$$

The first term contains the QSS drift while the second one represents the memory correction to this, which involve the memory function (4.37). Our derivation allows this memory to cover the behaviour around multiple fixed points of the system, due to its nonlinear dependence on $\boldsymbol{x}^{\mathrm{s}}$.

Our linearisation approach (4.19) implies that the memory term captures dynamical effects that are of first order in the deviations of the bulk network from its QSS. We will consider below an approximate self-consistent way of incorporating higher order corrections, which turns out also to simplify the numerical evaluation of the memory terms. To derive this self-consistent approach, consider the factor $f^0_{bs}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}),\tau)$ that

from (4.33,4.37) appears in the memory function $M_s(\boldsymbol{x}^{\mathrm{s}}, \tau)$. In the actual memory integral this is evaluated for $\boldsymbol{x}^{\mathrm{s}}(t')$ and $\tau = t - t'$, i.e. as $f^0_{bs}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}(t'), t - t')$. Similarly $M_s(\boldsymbol{x}^{\mathrm{s}}(t'), t - t')$ evolves the matrix exponential of

$$\int_0^{t-t'} d\tau' \; \boldsymbol{l}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}(t'), \tau')) = \int_{t'}^{t} dt'' \; \boldsymbol{l}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}(t'), t'' - t')) \tag{4.40}$$

As explained above, ϕ_v is the flow generated only by the QSS drift, i.e. without memory corrections. But the memory terms change the flow, so we can make the approach self-consistent by substituting for ϕ_v the actual time evolution *with* memory. This corresponds to replacing

$$\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}(t'), t - t') \to \boldsymbol{x}^{\mathrm{s}}(t) \tag{4.41}$$

as we are just propagating the subnetwork state from $\boldsymbol{x}^{\mathrm{s}}(t')$ by a time difference $t - t'$ to $\boldsymbol{x}^{\mathrm{s}}(t)$. Making this replacement also in the matrix exponential in (4.33) changes the memory term $\mathcal{M}_s(t) = \int_0^t dt' \; M_s(\boldsymbol{x}^{\mathrm{s}}(t'), t - t')$ into

$$\begin{aligned} \tilde{\mathcal{M}}_s(t) = \sum_{b''} \int_0^t dt' \; \sum_{b'} c_{b'}(\boldsymbol{x}^{\mathrm{s}}(t')) \left(e^{\int_{t'}^t dt'' \, \boldsymbol{l}(\boldsymbol{x}^{\mathrm{s}}(t''))} \right)_{b'b''} \\ \times f^0_{b''s}(\boldsymbol{x}^{\mathrm{s}}(t)) \end{aligned} \tag{4.42}$$

The dependence on the subnetwork species s on which the memory acts is contained only in the – now t'-independent – factor $f^0_{b'',s}(\boldsymbol{x}^{\mathrm{s}}(t))$. As shown in Appendix 4.B, the memory integrals in the first line can then be calculated efficiently as solutions to differential equations, one for each bulk species b''. Conceptually, on the other hand, the self-consistent memory term is more complicated. In the original formulation (4.39), the memory is a superposition of effects from all past times t': the state $\boldsymbol{x}^{\mathrm{s}}(t')$ of the subnetwork affects the behaviour of the bulk and feeds back into the subnetwork at time t. In (4.42), the way this feedback acts is additionally modulated by the entire time

evolution of the subnetwork between times t' and t. In the applications considered below both approaches yield very similar quantitative results so which one is chosen depends on one's aim: for numerical calculations of memory effects the self-consistent version is more efficient, while the memory functions themselves are easier to analyse in the original version because they depend – in addition to time difference, which always features – only on the subnetwork state at one time t'.

4.2.7. Memory decomposition

As in Chapter 2 we are able to do channel decomposition of the memory (original version, not self-consistent approximation), to understand the contribution of interactions within a network. We take advantage of the two partial derivative expressions in (4.46) to decompose the outgoing and incoming memory exactly. We begin by writing the expression for the memory function explicitly, combining Eqs. (4.20, 4.33, 4.37, 4.38):

$$
\begin{aligned}
M_s(\boldsymbol{x}^{\mathrm{s}},\tau) &= \sum_{b'}\sum_{s'}\sum_{b''}(\boldsymbol{J}^{-1})_{b'b''}\frac{\partial R_{b''}}{\partial x_{s'}}R_{s'}f_{b's}(\boldsymbol{x}^{\mathrm{s}},\tau) && (4.43)\\
&= \sum_{b'}\sum_{s'}\sum_{b'',c}(\boldsymbol{J}^{-1})_{b'b''}\frac{\partial R_{b''}}{\partial x_{s'}}R_{s'}\sum_{c}E_{b'c}(0,\tau)f^{0}_{cs}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}},\tau)) && (4.44)\\
&= \sum_{b'}\sum_{s'}\sum_{b''}(\boldsymbol{J}^{-1})_{b'b''}\frac{\partial R_{b''}}{\partial x_{s'}}R_{s'}\sum_{c}E_{b'c}(0,\tau)\frac{\partial R_s}{\partial x_c}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}},\tau)) && (4.45)
\end{aligned}
$$

where the first three factors are evaluated at $\boldsymbol{x}^{\mathrm{s}}$. Swap index labels and reorder factors to make this more intelligible:

$$
\begin{aligned}
M_s(\boldsymbol{x}^{\mathrm{s}},\tau) &= \sum_{b's'b''b}(\boldsymbol{J}^{-1})_{b''b'}\frac{\partial R_{b'}}{\partial x_{s'}}R_{s'}E_{b''b}(0,\tau)\frac{\partial R_s}{\partial x_b}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}},\tau)) && (4.46)\\
&= \sum_{s'}\sum_{bb'}\left(\sum_{b''}\frac{\partial R_s}{\partial x_b}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}},\tau))E_{b''b}(0,\tau)(\boldsymbol{J}^{-1})_{b''b'}(\boldsymbol{x}^{\mathrm{s}})\frac{\partial R_{b'}}{\partial x_{s'}}(\boldsymbol{x}^{\mathrm{s}})R_{s'}(\boldsymbol{x}^{\mathrm{s}})\right)
\end{aligned}
$$

The interpretation of this is as follows. A small change of subnetwork species s', $R_{s'}dt'$, in a small time interval dt' in the past first feeds through to a change of bulk species

b'. Only the bulk species b' whose time evolution depends explicitly on s' contribute here because of the factor $\partial R_{b'}/\partial x_{s'}$, so the b' can be interpreted as *outgoing channels* for the signal from s'. After propagation in the bulk network the signal returns via another bulk species. Here only bulk species b contribute that appear explicitly in the time evolution of subnetwork species s as indicated by the factors $\partial R_s/\partial x_b$. The b can therefore be interpreted as *incoming channels*. Overall, we have memory effects from s' onto s, via an outgoing channel (s' to b') and an incoming channel (b to s). Consistent with this interpretation, the outgoing channel "susceptibilities" $\partial R_{b'}/\partial x_{s'}$ are evaluated for the past, i.e. "sending", state $\boldsymbol{x}^{\mathrm{s}} \equiv \boldsymbol{x}^{\mathrm{s}}(t')$ of the subnetwork. The incoming channel susceptibilities $\partial R_s/\partial x_b$, on the other hand, are evaluated at the current time t as shown by the propagation via $\boldsymbol{\phi}_v$ across the time difference $\tau = t - t'$. Indeed, within the self-consistent approximation (4.69), this propagation corresponds directly to evaluation at the current state $\boldsymbol{x}^{\mathrm{s}}(t)$. The self-consistent approximation follows a similar construction and thus we can also perform a channel decomposition there (Appendix 4.C).

4.3. Dynamical Applications

We test the effectiveness of our method on systems that contain multiple steady states, oscillatory behaviours and complex transient dynamics. The systems we have chosen are relevant to a wide range of biological contexts.

4.3.1. Multistability

We use the method on a series of multistable systems defined by mutually repressive Hill functions with an "or" logic.:

$$\frac{d}{dt}x_j = \frac{a}{1+\sum_{i\neq j} x_i^n} - x_j \tag{4.47}$$

The above equation constitutes an "or" logic because of the sum of the terms in the denominator, where even if only one repressor has a high concentration, the production rate will become very low.

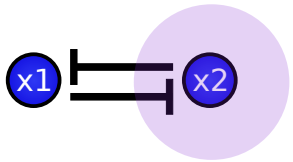


Figure 4.1: Network illustrating the bistable switch system defined using cross-repressive Hill functions, see (4.47). Blunt arrows indicate repression, purple shading indicates the species placed in the bulk.

We test the method on the simplest case with two nodes $\{x_1, x_2\}$. We place x_2 in the bulk and calculate the memory function for the single remaining subnetwork species x_1. This depends on the past concentration $x_1(t')$ and the time difference $\tau = t - t'$ (Fig. 4.2). We observe that the memory becomes zero at each fixed point as expected from Chapter 2, where the memory was obtained as an expansion (to quadratic order) in deviations of $\boldsymbol{x}^{\mathrm{s}}$ from a fixed point. To leading order the memory grows linearly with this deviation, and in line with this we see it changing sign at every fixed point. The sign of the memory in all cases is opposite to that of the drift, so the memory delays the relaxation time to the corresponding steady state. This makes intuitive sense as in the original system, the bulk species' state changes slowly, as opposed to the QSS system.

To test the accuracy of our method in capturing the transient temporal dynamics we set the initial condition of x_1 to be close to the saddle point of the QSS dynamics; here we are furthest from the stable fixed points and so can test the limits of the method. As a ground truth baseline we choose the full dynamics of the original system, setting x_2 at time zero to its QSS value with respect to the value of x_1. We compare this with the subnetwork dynamics predicted by the simple QSS dynamics, and by our approach, which includes memory corrections. For this example we evaluate both the

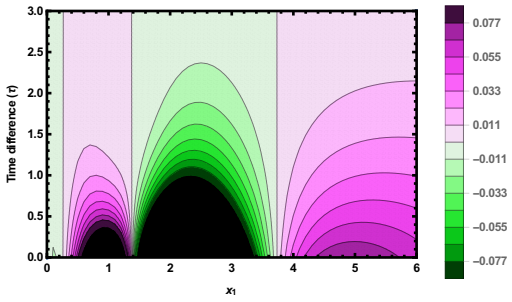


Figure 4.2: Memory function for the bistable switch of Fig. 4.1; values go from negative to positive as indicated by the scale bar on the right. Black indicates regions where the values lie outside the range of this scale bar.

vanilla memory description (4.39), which we label ZMv (Zwanzig-Mori vanilla), and the self-consistent memory (4.42), which we abbreviate ZMs (Zwanzig-Mori self-consistent). We find that both replicate the behaviour of the original system very well, independently of whether the initial condition leads to the low- or high-x_1 fixed point eventually. The QSS approximation, on the other hand, reaches the steady state unrealistically fast (Fig. 4.3). Given that the ZMs is substantially easier to implement for time course prediction, we proceed with that approach from this point onward (Appendix 4.B), naturally for plotting memory functions we use the original method.

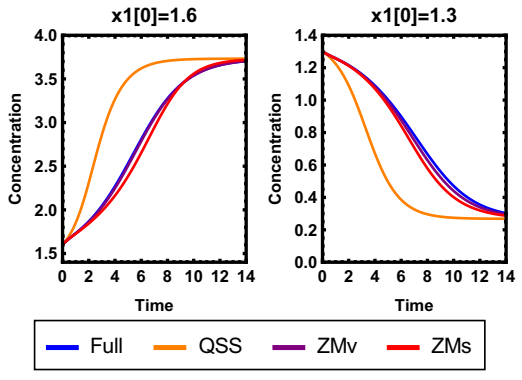


Figure 4.3: Time courses of bistable switch for $a = 4$ and $n = 2$. These demonstrate the capacity of the subnetwork equation with memory effects to capture the timescale and shape of transients reaching either of the two stable fixed points. The QSS approximation significantly underestimates the length of the transients.

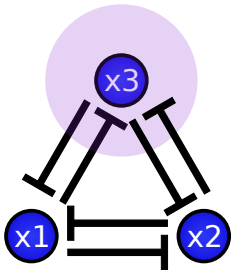


Figure 4.4: Network illustrating the tetrastable system used. Blunt arrows indicate repressions, purple shading identifies the species placed in the bulk.

We further test this on a tetrastable system defined in the same way as (4.47) with variables $\{x_1, x_2, x_3\}$. We consider the subnetwork containing x_1 and x_2, which will allow us to investigate the effect of the memory effects on the shapes of the basins of attraction of the different (stable) fixed points. For the parameter values we use there are four such fixed points for the full network: three where only one species has high concentration and the other two low, and one where all concentrations are equal (Fig. 4.4).

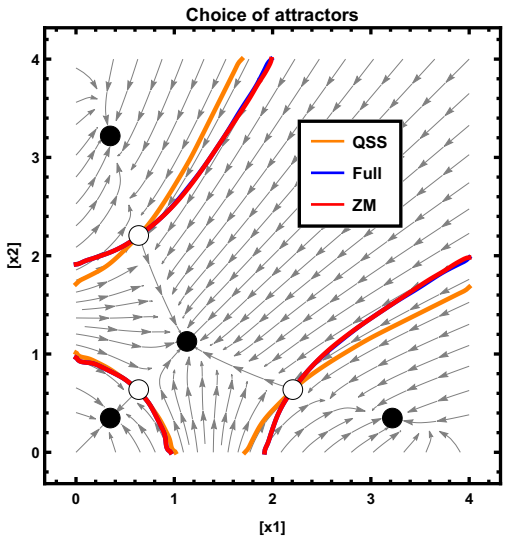


Figure 4.5: Basins of attraction of the four stable fixed points (black dots) of the system in Fig. 4.4, for the full dynamics (blue), the QSS approximation (orange) and the subnetwork equations with memory (ZMs, red). Note that the ZMs boundaries are almost indistinguishable from the those for the full dynamics; stream plots are shown for the QSS approximation.

The boundaries of the basins of attraction can be viewed as the points where fate decisions of the system change. We find that the QSS system fails to replicate the the same decision process as the original system, whereas the ZMs accurately chooses the same steady state (Fig. 4.5) *and* the timing to get to the steady state, as shown in Fig. 4.3.

4.3.2. Oscillations

We further explored the ability of the subnetwork equations with memory to reproduce oscillations arising from a uni-directional repressive network. We used the well documented repressilator system (Elowitz and Leibier, 2000), with variables $\{x_1, x_2, x_3\}$ and repressive interactions as shown in Fig. 4.6 and represented mathematically by

$$\frac{d}{dt}x_j = \frac{a}{1 + x_{j-1}^n} - x_j \tag{4.48}$$

where $x_0 \equiv x_3$.

We first explored the bifurcation diagram that results from varying both system parameters a and n, in order to see whether the subnetwork equations with memory can replicate the corresponding 2D Hopf bifurcation of the original system, from damped to sustained oscillations (Fig. 4.7).

While the QSS system is unable to generate sustained oscillations at all, we find that the projection correctly replicates the existence of sustained oscillations and predicts a qualitatively correct bifurcation diagram. The period and amplitude of sustained oscillations in the relevant parameter regime is less well replicated (Data not shown). For damped oscillations the subnetwork equations with memory work very accurately in predicting the full temporal dynamics as observed in Fig. 4.8. By contrast, the QSS displays almost no oscillatory behaviour, highlighting the importance of the memory effects.

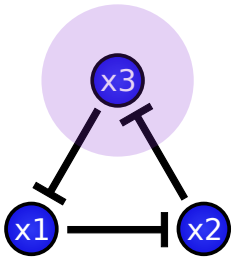


Figure 4.6: Network illustrating the repressilator system. Blunt arrows indicate repressions, purple shading labels the species placed in the bulk.

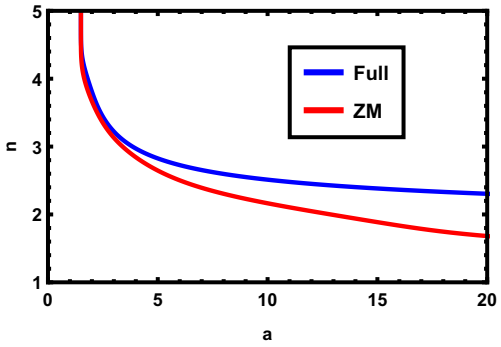


Figure 4.7: Bifurcation diagram of the repressilator in terms of system parameters a and n from (4.48). The lines represent super-critical Hopf bifurcations, with damped oscillations occurring to the left and below the curves and sustained oscillations above and to the right.

In order to understand in more detail how memory effects generate oscillations, we go on to analyse the corresponding memory functions. We first plot the memory amplitude, i.e. the value $M(\boldsymbol{x}^{\mathrm{s}}, 0)$ for memory from the immediate past ($t' = t$) across the configuration space of our subnetwork and observe two very distinct regions with positive and negative memory amplitude, separated by a line where the memory amplitude vanishes (black). When plotting the time course from Fig. 4.8 in the same representation we observe that it crosses the black line many times. The memory amplitude changes sign approximately at the same time and so is able to drive the oscillations (Fig. 4.9).

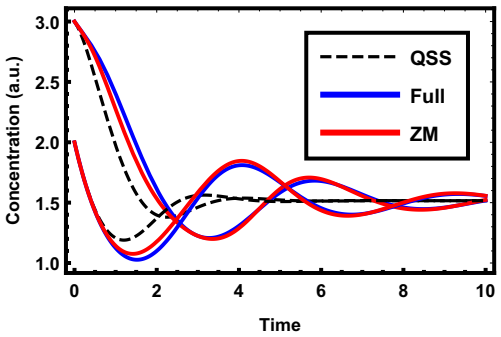


Figure 4.8: Damped oscillations can accurately be reproduced by the projection approach while the QSS fails to replicate both the timing and the amplitude of the oscillations.

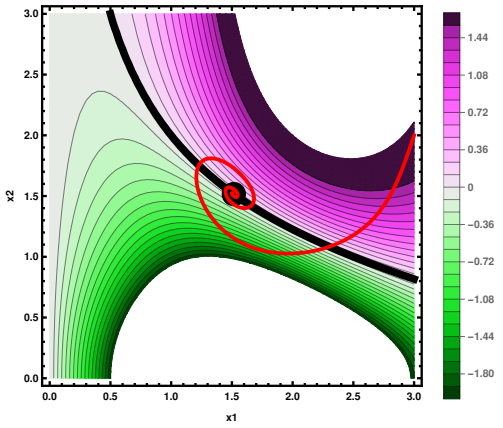


Figure 4.9: Memory amplitude at $\tau = 0$, which captures the changes in memory amplitude from negative to positive, the thick black line indicates where the memory value is zero. The path from Fig. 4.8 is represented by the red curve.

4.4. Neural tube network (transients and multistability)

We go on to apply the projection to a biologically relevant system with several bifurcations and non-trivial dynamical properties, specifically the neural tube network described in (Cohen et al., 2014). As a first approach, similarly to how we proceeded in Chapter 3, we place Nkx2.2 and Olig2 in the subnetwork given that these two alone generate a bistable switch, so the remaining bulk behaviour is now provided by the two other species (Irx3 and Pax6).

We test the method in the position along the neural tube where the model has the

most complexity, a region of known tristability ($p = 0.65$, see Chapter 3), and compare it with the original system and the QSS approximation (Fig. 4.10). We find as in the tetrastable case, that the projection is very accurate for replicating the choice of steady state, in contrast to the QSS (Fig. 4.5 & 4.10). We test the method's accuracy at all other neural tube positions and always find a good match between the WT and projection approach, similar to that displayed in Fig. 4.10 (data not shown).

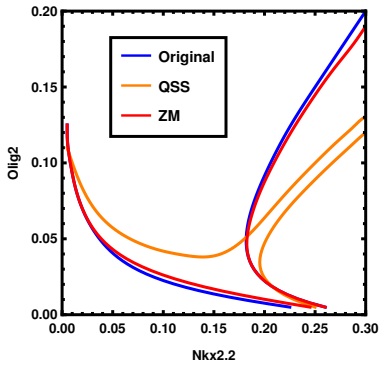


Figure 4.10: Taken at tristable neural tube position ($p = 0.65$), lines indicate boundaries between different fate choices. Possible steady states are p3 (high Nkx2.2), pMN (high Olig2 $\&$ medium Pax6) and p2 (High Irx3 $\&$ Pax6). Colours indicate boundaries for the different approximations to the dynamics (QSS and ZM) and the full dynamics.

We analyse the trajectory of the systems at all neural tube positions, using the experimentally determined initial condition for Nkx2.2 and Olig2 of zero, and compare the ZM with the original system and the QSS reduction. The ZM system shows a very good fit with the original system at all positions (Fig. 4.11 displays results for the most interesting case $p = 0.1$), demonstrating that the memory functions are capable of accurately capturing not just final cell fate decisions but also the timing to make such decisions. This behaviour is fundamental in neural tube patterning, leading to a wave of expression of Olig2 observed at a tissue scale.

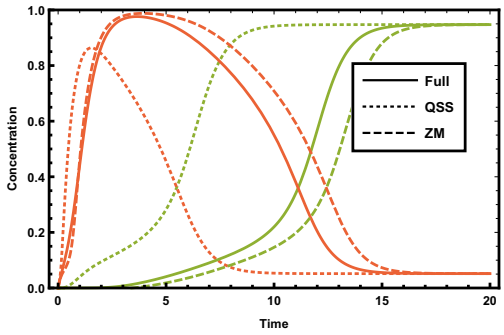


Figure 4.11: Neural tube dynamics in monostable region for Nkx2.2 ($p = 0.1$), with Nkx2.2 is labelled in green and Olig2 is labelled in red. A transient expression of Olig2 exists in the biological system and the original equations can replicate this behaviour. In contrast, the QSS approximation is unable to capture the long Olig2 transient while the ZM projection replicates the behaviour qualitatively well.

4.4.1. Linearised comparison

The Zwanzig-Mori method developed in this Chapter can be linearised around fixed points to perform a comparison with the method derived in Chapter 2. We compare the accuracy of both methods in relationship to the original system and find, as expected, that this depends greatly on the initial conditions of the bulk. This makes sense because the choice of bulk initial conditions constitutes the baseline assumption of each method. When the bulk species are initialised at the steady state, the method from Chapter 2 outperforms the current method as this is the assumption made for that derivation (Fig. 4.12). Conversely, the method derived in this chapter outperforms the previous method if the bulk is starting at the QSS corresponding to the subnetwork initial state as is to be expected (Fig. 4.12). We also highlight that the predictions of the methods are exact when close to the steady state and the "ideal" bulk initial conditions have been chosen as can be appreciated from Fig. 4.12. A further interesting take home message is that, regardless of which assumption is used, memory functions provide a more accurate behaviour than either memoryless system as can be seen from the longer relaxation times of the trajectories with memory in Fig. 4.12. This is of particular importance as in this system, like in many biological systems, we do not know the initial conditions of the

bulk; our results show that in spite of this we can predict dynamics that behaves in a way that is qualitatively similar to the full system.

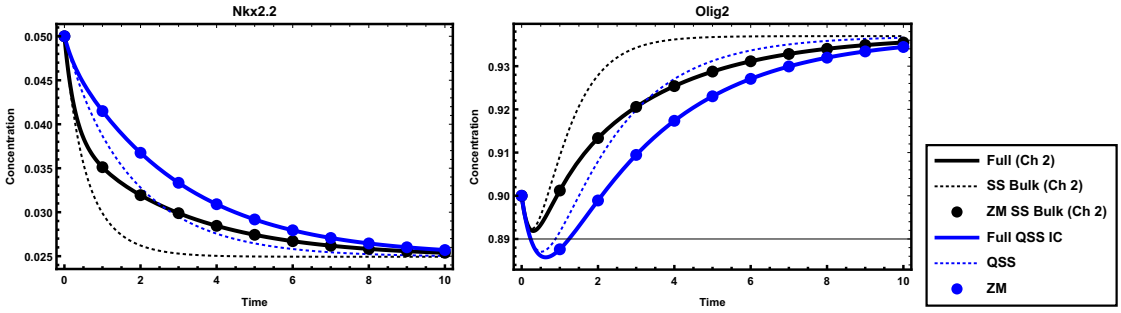


Figure 4.12: Linearised version of approaches from Chapter 2 and from this chapter for temporal trajectories of chosen subnetwork species Nkx2.2 (left plot) and Olig2 (right plot). "Full (Ch 2)": full linearised system with initial conditions for the bulk species at the calculated final steady state. "SS Bulk (Ch 2)": linearised system with only subnetwork terms, expanded around the calculated final steady state. "ZM SS Bulk (Ch 2)": linearised version of the system derived in Chapter 2. "Full - QSS IC": full linearised system with initial conditions at the quasi-steady state. "QSS": linearised system with only subnetwork terms expanding around QSS. "ZM": linearised version of the system derived in this chapter. The precision of each approach varies depending on the choice of initial bulk concentration, due to this being a fundamental assumption intrinsic to each method.

We next go on to perform a memory decomposition of the amplitude of the linearised memory as we did in Fig. 3.3, but now for the method derived in this chapter (Fig. 4.13). We find very similar profiles to those of Fig. 3.3. The results also highlight the relative weakness of the memory from Olig2 into Nkx2.2 via Pax6, supporting the results from Chapter 3. The method derived in this chapter, however, does not rely on an expansion near a steady state so we next go on to search for a decomposition of the full memory terms, now that we know they can provide insightful information.

4.4.2. Decomposing nonlinear memory functions

We seek to understand the structure of the *full* memory functions and how it relates to the observable dynamics. We perform our memory decomposition approach across all neural tube positions, and find dramatic differences in memory channels, in sign of the

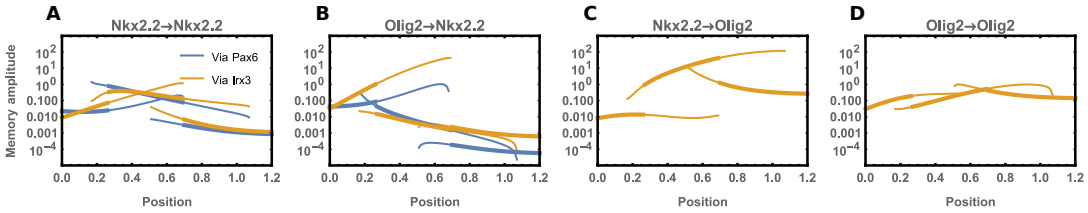


Figure 4.13: Linear memory amplitude decomposition. Analogue of Fig. 3.3 for the nonlinear memory functions developed in this Chapter, linearised around fixed points. (A) Amplitude of memory (memory function at $\Delta t = 0$) of Nkx2.2 to itself along the neural tube. There are multiple lines as the analysis was performed at all possible stable steady states. The vertical axis is logarithmic to make the range of amplitudes easier to appreciate. Colours identify the memory amplitude contribution from the two possible bulk channels, via Irx3 and Pax6, respectively. Thick lines indicate physiological states, while thin lines indicate states that are not usually observed in vivo. (B) Linear memory amplitude of (past) Olig2 on Nkx2.2 along the neural tube. The memory via Pax6 is for the most part below the memory via Irx3, in each pair of corresponding curves. (C,D) Memory amplitudes of Olig2 to Nkx2.2 (C) and to itself (D). No channel decomposition is performed as Olig2 receives memory only via the Irx3 channel.

memory, and also with respect to neural tube position (Fig. 4.15). The decomposition is performed at all positions, though only some positions are shown as illustrative examples (Fig. 4.15). In order to find the contributions of each channel we integrate a small time difference from $\tau = 0$ to $\tau = 0.1$. We then divide the resulting value by the absolute value of the drift of the memoryless system. The reasoning for this is that memory functions only make a difference to the dynamics if they are of a comparable or larger magnitude than the drift of the memoryless system. Since we are primarily interested in the magnitude of the memory contributions we take the absolute value of the results and plot them on a log scale (Fig. 4.16). We find that the largest contributor to memory in the system is signal from Olig2 to Nkx2.2 through Irx3, suggesting the information stored through this route is fundamental for the transient behaviour.

By further observing our decomposition we find which memory channels have the least impact on the temporal trajectory based on their contribution (Fig. 4.16). The resulting channels are all the ones from Olig2 into Nkx2.2 via Pax6. Based on our analysis of the memory in the entire phase space we can now discuss why this is the case. In

most situations the contribution from Irx3 is much larger than from Pax6, making the effects of Pax6 negligible (Fig. 4.16). There are regions where the contribution of Pax6 is greater than Irx3 but these are regions of phase space that the system never explores in biologically relevant situations (coexpression of Nkx2.2 and Olig2) (Fig. 4.16). This all fits in with the results from Chapter 3.

Memory decomposition allows us to run temporary trajectories while removing specific channels (Appendix 4.3) so we proceed to remove the channels identified as weak above, therefore effectively setting them to QSS. As expected we find little impact on the trajectories (for illustrative position $p = 0.1$ see Fig. 4.14). The same link was removed in Chapter 3, but the interpretation of what the weak impact of the removal means has to be considered carefully. There are two possibilities for the origin of a low memory strength of a channel: the first is that the link encoding the memory is generally weak and therefore the link itself will contribute very little to the overall dynamics. The other is that, as in our case with the repression of Pax6 from Olig2 leading to a de-repression of Nkx2.2, happens essentially instantly without a temporal delay *i.e.* at QSS. In biological terms this can be loosely interpreted as a positive activation of Nkx2.2 by Olig2.

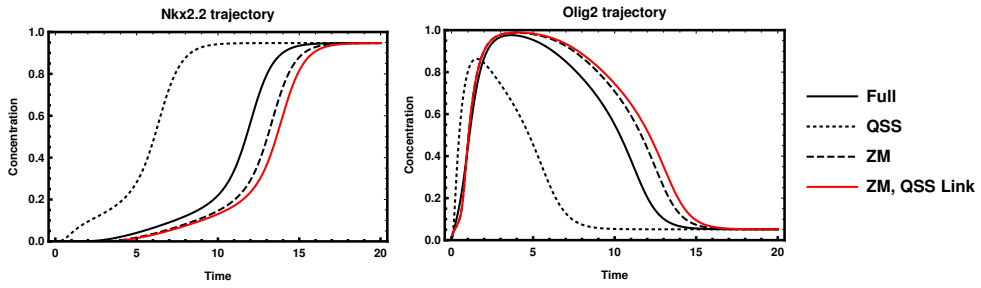


Figure 4.14: Plots of concentration dynamics of Nkx2.2 (left) and Olig2 (right) at $p = 0.1$. Full dynamics; QSS and ZM are plotted as labelled in Fig. 4.11. In red is shown the result of setting the inhibition of Olig2 to Pax6 to be at QSS, there is little change between this and the ZM projection indicating little effect of this memory channel.

When combined with the results from Chapter 3 we can determine what is happening at different neural tube positions regarding the above mentioned link. At neural tube

positions dorsal to approximately 0.2 the link of Olig2 repression of Pax6 has little effect on the dynamics of the subnetwork species in general, *not just* near steady state. Ventral to the position 0.2 this link is only weak near the Nkx2.2 steady state but otherwise does contribute to the dynamics. In this case the reason that the effect of the memory function in Fig. 4.14 is low is that the link is at QSS, meaning that the information from Olig2 into Nkx2.2 through Pax6 is transmitted rather fast compared to the typical timescale of concentration changes. By combining the results from Chapter 3 and this chapter, we are thus able to explain the importance of a specific link, and how its importance varies across neural tube position.

4.5. Summary

In summary we have derived a general dimensionality reduction technique by solving non-trivial memory functions derived from the work of Alexandre Chorin and collaborators. We have shown its high precision in systems where we do not use any prior knowledge of the dynamics or steady states of the systems. The methodology relies on a baseline assumption of a QSS bulk, which is then corrected for by the presence of memory functions. We have shown that the corrections provided by the memory functions result in very accurate dynamics in a variety of systems with different properties. By construction the reduced system is able to exactly recapitulate all stable fixed points of a system, because of our assumption that the bulk is chosen to always have a unique QSS. We show that, beyond this, the method is able to choose the correct fixed point from a wide range of initial conditions, predicting fate decisions much more accurately than the memoryless case. We also demonstrate the system's ability to capture temporal behaviour such as sustained or dampened oscillations and accurately recreate the corresponding bifurcation at similar parameter changes to the original system.

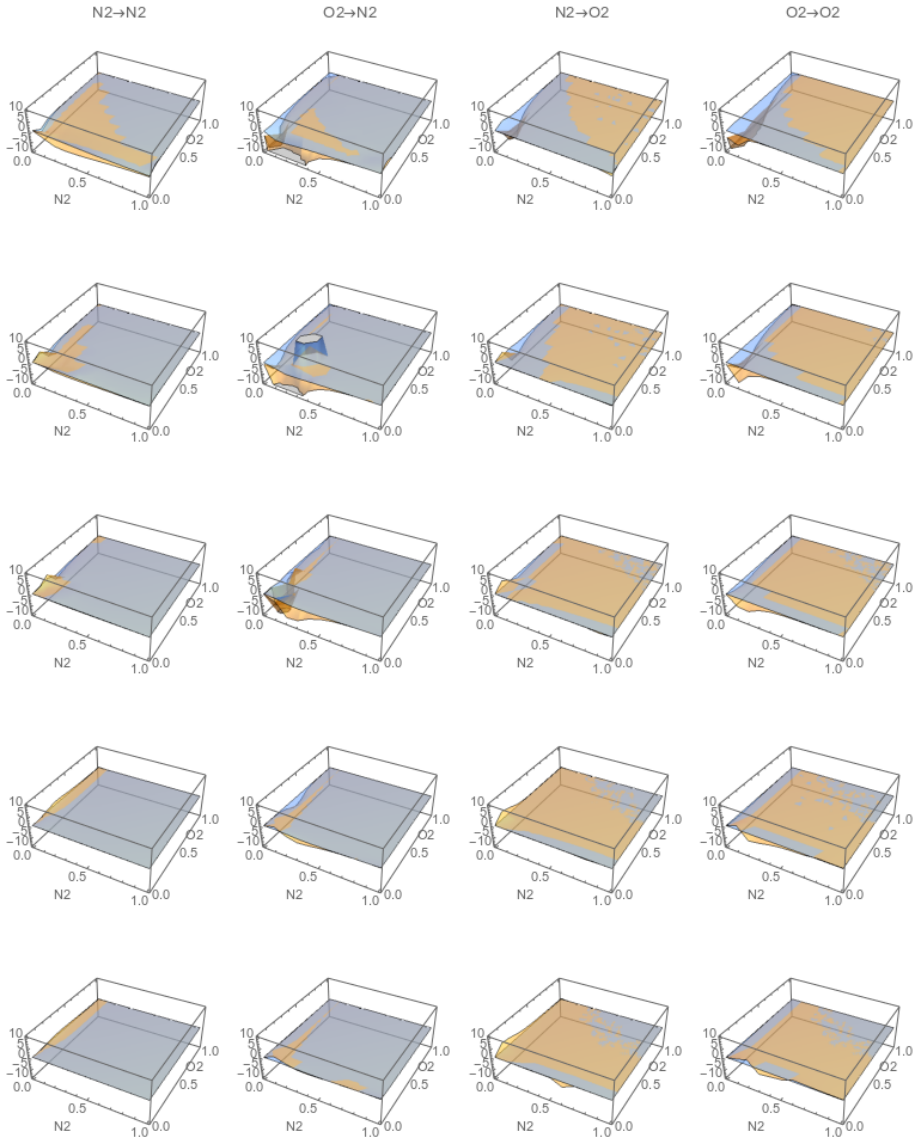


Figure 4.15: Memory decomposition at all neural tube positions and all phase space. Each column represents memory coming from a particular subnetwork TF into another subnetwork TF as indicated by the labels at the top. Each row is a different neural tube position p, starting at $p = 0.1$ at the top and increasing in steps of 0.2. Blue indicates memory through Pax6, yellow through Irx3; due to the structure of the GRN there is no crosstalk between bulk TFs. Values have been calculated as an integration of a small timestep from $\tau = 0$ to $\tau = 0.1$ and divided by an equivalent value from the drift to focus on areas where memory can have a substantial effect on the trajectories.

We apply the method to the neural tube network and find results that complement those of Chapter 3 regarding the inhibition of Pax6 by Olig2. We are able to understand the reason this link presents a different behaviour that results in a loss of robustness

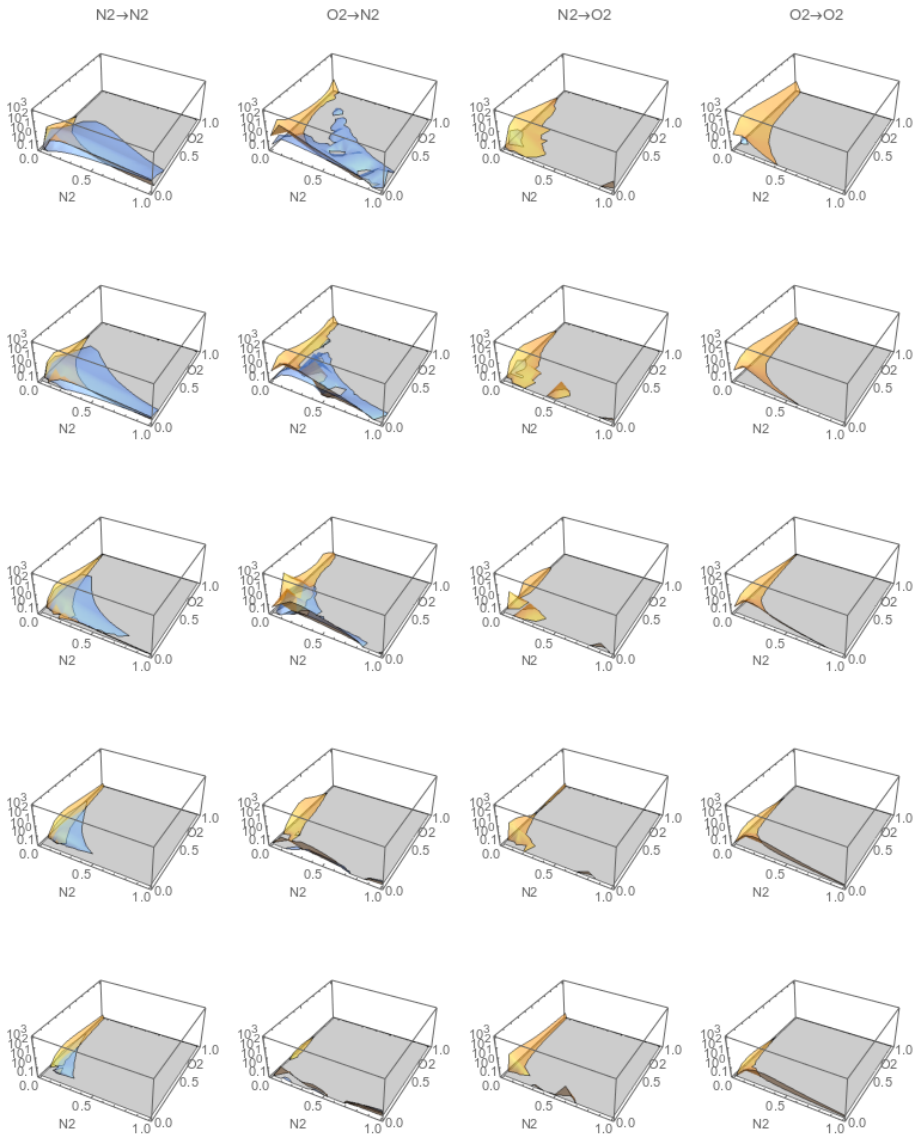


Figure 4.16: Equivalent plot to Fig. 4.15 but with absolute value of the memory and drift. The y axis is now chosen as logarithmic to show the full range.

to initial conditions when removed. We highlight the complementarity of the approach derived in this chapter with that of Chapter 2; together they can be used to determine the relative strength of an interaction as well as the importance of such a link producing a temporal delay in propagating signal or not.

4.A. Appendix: Solution for f

In this appendix we will show that the solution to the partial differential equation (4.28) for $f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau)$ is given by (4.33). For the sake of brevity we will introduce some abbreviated notation for the required matrix exponentials, as follows:

$$\boldsymbol{E}(\boldsymbol{x}^{\mathrm{s}}, \tau_1, \tau_2) = e^{\int_{\tau_1}^{\tau_2} d\tau\, \boldsymbol{l}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}, \tau))} \tag{4.49}$$

Because of the time integral in the exponent we have to use time ordering. We will adopt a left-to-right time order convention throughout, with the latest time appearing as the rightmost factor, so that the first terms in the Taylor series for $\boldsymbol{E}$ read

$$\begin{aligned}\boldsymbol{E}(\boldsymbol{x}^{\mathrm{s}}, \tau_1, \tau_2) =& \mathbf{1} + \int_{\tau_1}^{\tau_2} d\tau'\, \boldsymbol{l}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}, \tau')) \\ &+ \int_{\tau_1}^{\tau_2} d\tau' \int_{\tau'}^{\tau_2} d\tau''\, \boldsymbol{l}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}, \tau'))\boldsymbol{l}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}, \tau'')) + \dots\end{aligned} \tag{4.50}$$

with $\mathbf{1}$ the identity matrix. We will generally omit the $\boldsymbol{x}^{\mathrm{s}}$ argument from $\boldsymbol{E}$ where it is clear from the context. The proposed solution (4.33) for $f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau)$ is of the form

$$f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) = \sum_{b'} E_{bb'}(0, \tau) f^0_{b's}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}, \tau)) \tag{4.51}$$

Here $f^0_{bs}(\cdot)$ is the initial condition $f_{bs}(\cdot, 0)$, which is given explicitly in (4.20). Further, as discussed in the main text, $\boldsymbol{\phi}_v$ is the flow with only the QSS drift $\boldsymbol{v}(\boldsymbol{x}^{\mathrm{s}})$ and without memory effects. This flow is defined for any $\boldsymbol{x}^{\mathrm{s}}$ as the solution of the differential equation

$$\frac{\partial}{\partial \tau}\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}, \tau) = \boldsymbol{v}(\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}, \tau)), \qquad \boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}, 0) = \boldsymbol{x}^{\mathrm{s}} \tag{4.52}$$

where as before $\boldsymbol{v}(\boldsymbol{x}^{\mathrm{s}}) = \boldsymbol{R}^{\mathrm{s}}(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*}(\boldsymbol{x}^{\mathrm{s}}))$ We can then write the time evolution of $\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau)$ as:

$$\frac{\partial}{\partial \tau}\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau) = R_s(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau), \boldsymbol{x}^{\mathrm{b}*}) \tag{4.53}$$

Following the same argument as in the main text for the full time evolution $\boldsymbol{\phi}(\boldsymbol{x}, t)$, the time evolution $(\partial/\partial\tau)\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau)$ can then equivalently be written as

$$\frac{\partial}{\partial \tau}\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau) = \sum_{s'} R_{s'} \frac{\partial}{\partial x_{s'}}\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau) \tag{4.54}$$

with $R_{s'}$ evaluated as usual at $(\boldsymbol{x}^{\mathrm{s}}, \boldsymbol{x}^{\mathrm{b}*}(\boldsymbol{x}^{\mathrm{s}}))$. It is important to note that $\boldsymbol{\phi}_v(\boldsymbol{x}^{\mathrm{s}}, \tau)$ is different from $\boldsymbol{\phi}(\boldsymbol{x}, t)$, with the former describing only the subnetwork dynamics assuming a bulk at QSS while the latter encodes the dynamics of the full network. Nonetheless, as we will only have to deal with $\boldsymbol{\phi}_v$ in the following, we will drop the "0" subscript for brevity.

After these preparations we can move on to the task of proving that our proposed form of f correctly solves (4.28). We will use c and d as bulk indexes, similar to b'. We insert our solution (4.51) into the LHS of (4.28) to obtain

$$\begin{aligned}\frac{\partial}{\partial \tau} f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) = &\sum_{b'c} E_{bc}(0, \tau) l_{cb'}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau)) f^0_{b's}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau)) \\ &+ \sum_{b's'} E_{bb'}(0, \tau) \frac{\partial}{\partial \tau}\phi_{s'}(\boldsymbol{x}^{\mathrm{s}}, \tau) \frac{\partial}{\partial \phi_{s'}} f^0_{b's}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau))\end{aligned} \tag{4.55}$$

with the terms in the first and second line coming from the τ-dependence of $f^0_{b's}$ and $E_{bb'}$, respectively. On the RHS of (4.28) we begin by evaluating the second term, which

contains the part of the Liouvillian acting on subnetwork concentrations:

$$\sum_{s'} R_{s'} \frac{\partial}{\partial x_{s'}} f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) = \sum_{s'b'cd} \int_0^\tau d\tau' E_{bc}(0,\tau') R_{s'} \frac{\partial}{\partial x_{s'}} l_{cd}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau')) \tag{4.56}$$
$$\times E_{db'}(\tau', \tau) f^0_{b's}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau))$$
$$+ \sum_{b'} E_{bb'}(0,\tau) \sum_{s's''} R_{s'} \frac{\partial}{\partial x_{s'}} \phi_{s''}(\boldsymbol{x}^{\mathrm{s}}, \tau) \frac{\partial}{\partial \phi_{s''}} f^0_{b's}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau))$$

We can then use (4.54) to obtain the following equivalence

$$\sum_{s'} R_{s'} \frac{\partial}{\partial x_{s'}} l_{cd}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau')) = \sum_{s's''} \frac{\partial}{\partial \phi_{s''}} l_{cd}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau')) R_{s'} \frac{\partial}{\partial x_{s'}} \phi_{s''}(\boldsymbol{x}^{\mathrm{s}}, \tau') \tag{4.57}$$
$$= \sum_{s''} \frac{\partial}{\partial \phi_{s''}} l_{cd}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau')) \frac{\partial}{\partial \tau'} \phi_{s''}(\boldsymbol{x}^{\mathrm{s}}, \tau') \tag{4.58}$$
$$= \frac{\partial}{\partial \tau'} l_{cd}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau')) \tag{4.59}$$

Using (4.54) also in the second line of (4.56), the latter becomes

$$\sum_{s'} R_{s'} \frac{\partial}{\partial x_{s'}} f_{bs}(\boldsymbol{x}^{\mathrm{s}}, \tau) = \sum_{b'cd} \int_0^\tau d\tau' \; E_{bc}(0,\tau') \left(\frac{\partial}{\partial \tau'} l_{cd}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau')) \right) \tag{4.60}$$
$$\times E_{db'}(\tau', \tau) f^0_{b's}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau))$$
$$+ \sum_{s''b'} E_{bb'}(0,\tau) \frac{\partial}{\partial \tau} \phi_{s''}(\boldsymbol{x}^{\mathrm{s}}, \tau) \frac{\partial}{\partial \phi_{s''}} f^0_{b's}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau))$$

The integrand in the first line is actually a complete τ'-derivative, as we now show. We start by differentiating the following product, with$\boldsymbol{l}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}}, \tau'))$ abbreviated as simply $\boldsymbol{l}(\tau')$:

$$\frac{\partial}{\partial \tau'} \boldsymbol{E}(0,\tau') \boldsymbol{l}(\tau') \boldsymbol{E}(\tau', \tau) = [\boldsymbol{E}(0,\tau') \boldsymbol{l}(\tau')] \boldsymbol{l}(\tau') \boldsymbol{E}(\tau', \tau) \tag{4.61}$$
$$- \boldsymbol{E}(0,\tau') \boldsymbol{l}(\tau') [\boldsymbol{l}(\tau') \boldsymbol{E}(\tau', \tau)]$$
$$+ \boldsymbol{E}(0,\tau') \left(\frac{\partial}{\partial \tau'} \boldsymbol{l}(\tau') \right) \boldsymbol{E}(\tau', \tau)$$

We observe that the first two terms cancel. Integrating between 0 and t thus gives the identity

$$\begin{aligned}\int_0^\tau d\tau'\ \boldsymbol{E}(0,\tau')\left(\frac{\partial}{\partial\tau'}\boldsymbol{l}(\tau')\right)\boldsymbol{E}(\tau',\tau) =&\boldsymbol{E}(0,\tau')\boldsymbol{l}(\tau')\boldsymbol{E}(\tau',\tau)\Big|_{\tau'=0}^{\tau'=\tau} \\ =&\boldsymbol{E}(0,\tau)\boldsymbol{l}(\tau)-\boldsymbol{l}(0)\boldsymbol{E}(0,\tau)\end{aligned} \tag{4.62}$$

Except for the last factor this is the first line of (4.60), which becomes

$$\begin{aligned}\sum_{s'}R_{s'}\frac{\partial}{\partial x_{s'}}f_{bs}(\boldsymbol{x}^{\mathrm{s}},\tau) =&\sum_{b'c}E_{bc}(0,\tau)l_{cb'}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}},\tau))f^0_{b's}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}},\tau)) \\ &-\sum_{b'c}l_{bc}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}},0))E_{cb'}(0,\tau)f^0_{b's}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}},\tau)) \\ &+\sum_{b's''}E_{bb'}(0,\tau)\frac{\partial}{\partial\tau}\phi_{s''}(\boldsymbol{x}^{\mathrm{s}},\tau)\frac{\partial}{\partial\phi_{s''}}f^0_{b's}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}},\tau))\end{aligned} \tag{4.63}$$

The first and last term together just give (4.55). The last two factors of the second term are seen from (4.51) to be identical to $f_{cs}(\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{s}},\tau))$, yielding overall

$$\sum_{s'}R_{s'}\frac{\partial}{\partial x_{s'}}f_{bs}(\boldsymbol{x}^{\mathrm{s}},\tau)=\frac{\partial}{\partial\tau}f_{bs}(\boldsymbol{x}^{\mathrm{s}},\tau)-\sum_{b'}l_{bb'}(\boldsymbol{x}^{\mathrm{s}})f_{b's}(\boldsymbol{x}^{\mathrm{s}},\tau) \tag{4.64}$$

This shows that the proposed solution (4.51) does indeed satisfy the differential equation (4.28) we wanted to solve.

4.B. Appendix: Mapping of self-consistent memory to differential equations

We show in this appendix how to map the subnetwork equations with self-consistent memory,

$$\frac{\partial}{\partial t}x_s = v_s(\boldsymbol{x}^{\mathrm{s}}(t)) + \tilde{\mathcal{M}}_s(t) \tag{4.65}$$

to a set of differential equations. The self-consistent memory term $\tilde{\mathcal{M}}_s(t)$ is given by (4.42)

$$\tilde{\mathcal{M}}_s(t) = \sum_{b''} \int_0^t dt' \ \sum_{b'} c_{b'}(\boldsymbol{x}^{\mathrm{s}}(t')) \left(e^{\int_{t'}^t dt'' \boldsymbol{l}(\boldsymbol{x}^{\mathrm{s}}(t''))} \right)_{b'b''} f^0_{b''s}(\boldsymbol{x}^{\mathrm{s}}(t)) \tag{4.66}$$

so can be written as

$$\tilde{\mathcal{M}}_s(t) = \sum_{b''} m_b(t) f^0_{bs}(\boldsymbol{x}^{\mathrm{s}}(t)) \tag{4.67}$$

with

$$m_b(t) = \int_0^t dt' \ \sum_{b'} c_{b'}(\boldsymbol{x}^{\mathrm{s}}(t')) \left(e^{\int_{t'}^t dt'' \boldsymbol{l}(\boldsymbol{x}^{\mathrm{s}}(t''))} \right)_{b'b} \tag{4.68}$$

It is then straightforward to check that

$$\frac{d}{dt} m_b(t) = c_b(\boldsymbol{x}^{\mathrm{s}}(t)) + \sum_{b'} m_{b'}(t) l_{b'b}(\boldsymbol{x}^{\mathrm{s}}(t)) \tag{4.69}$$

where the second term arises from the t-dependence of the matrix exponential. The $m_b(t)$ can therefore be obtained numerically by integrating the differential equations

(4.69) together with the subnetwork equations with (self-consistent) memory

$$\frac{d}{dt}x_s(t) = v_s(\boldsymbol{x}^{\mathrm{s}}(t)) + \sum_{b'} m_{b'}(t) f^0_{sb'}(\boldsymbol{x}^{\mathrm{s}}(t)) \tag{4.70}$$

Here the initial condition for the auxiliary variables is $m_b(0) = 0$.

4.C. Appendix: Self-consistent channel decomposition

The channel decomposition of Sec. 4.B can also be applied to the self-consistent memory approximation, as we now outline. Writing out the self-consistent memory term (4.66) explicitly and reordering and relabelling terms as in (4.46) gives

$$\begin{aligned}\tilde{\mathcal{M}}_s(t) &= \int_0^t dt' \sum_{s'}\sum_{bb'}\sum_{b''} \frac{\partial R_s}{\partial x_b}(\boldsymbol{x}^{\mathrm{s}}(t)) \left(e^{\int_{t'}^t dt'' \boldsymbol{l}(\boldsymbol{x}^{\mathrm{s}}(t''))}\right)_{b''b} \\ &\quad \times (\boldsymbol{J}^{-1})_{b''b'}(\boldsymbol{x}^{\mathrm{s}}(t')) \frac{\partial R_{b'}}{\partial x_{s'}}(\boldsymbol{x}^{\mathrm{s}}(t')) R_{s'}(\boldsymbol{x}^{\mathrm{s}}(t')) \\ &= \sum_{s'}\sum_{bb'} \frac{\partial R_s}{\partial x_b}(\boldsymbol{x}^{\mathrm{s}}(t)) m_{sbb's'}(t)\end{aligned} \tag{4.71}$$

where

$$m_{sbb's'}(t) = \int_0^t dt' \sum_{b''} (\boldsymbol{J}^{-1})_{b''b'}(\boldsymbol{x}^{\mathrm{s}}(t')) \frac{\partial R_{b'}}{\partial x_{s'}}(\boldsymbol{x}^{\mathrm{s}}(t')) R_{s'}(\boldsymbol{x}^{\mathrm{s}}(t')) \left(e^{\int_{t'}^t dt'' \boldsymbol{l}(\boldsymbol{x}^{\mathrm{s}}(t''))}\right)_{b''b} \tag{4.72}$$

From this last representation it follows that the $m_{sbb's'}(t)$ vanish at $t = 0$ and obey the differential equations

$$\frac{d}{dt} m_{sbb's'}(t) = (\boldsymbol{J}^{-1})_{bb'}(\boldsymbol{x}^{\mathrm{s}}(t)) \frac{\partial R_{b'}}{\partial x_{s'}}(\boldsymbol{x}^{\mathrm{s}}(t)) R_{s'}(\boldsymbol{x}^{\mathrm{s}}(t)) + \sum_{b''} m_{sb''b's'}(t) l_{b''b}(\boldsymbol{x}^{\mathrm{s}}(t)) \tag{4.73}$$

So also the channel-decomposed memory can be calculated from differential equations. Of course one only needs to calculate the $m_{sbb's'}$ for combinations (sb) and $(b's')$ where the corresponding channel susceptibilities are nonzero.

5

Boundary sharpness explained by a GRN

In previous chapters we focused on understanding how multiple steady states are achieved, how boundary positions are set, and how long it takes for steady state dynamics to be achieved. In this chapter we focus on understanding how spatial domains of gene expression are delimited in a precise way. This is a fundamental question as these domains will go on to differentiate to perform different functions which need to be clearly delimited. To study this we used stochastic differential equations (SDEs) to understand the effect of heterogeneity. We motivate our analysis from experimental results and provide a unifying explanation to multiple mutant phenotypes as well as the WT phenotype. In this

chapter all experimental results are obtained by Katherine Exelby (Exelby et al., 2019) unless noted otherwise.

5.1. Pax6$^{-/-}$ embryos present reduced boundary sharpness

We use the WT boundary between p3 and pMN domains as a baseline of a "sharp" boundary, and then proceed to look at genotypic perturbations that produce a less precise boundary. From previous experimental results it was known that mouse embryos lacking a functional Pax6 protein reduce the precision of the boundary between the p3 and pMN domains (Fig. 5.1C) Balaskas et al. (2012), Briscoe et al. (2000), Ericson et al. (1997). Cells expressing either Nkx2.2 or Olig2 are intermixed at the boundary between these two domains, although similar to wild-type few cells co-expression the two TFs presenting very little coexpression of these TFs. This phenotype had been documented but not characterised, for this reason we calculated the boundary precision at this region of intermixing as detailed extensively in Appendix A.3. The analysis showed that between e9.0 and e10.5, the pMN-p3 boundary becomes wider over time in Pax6$^{-/-}$ embryos, which can be clearly observed as a loss of precision (Fig. 5.1E & Fig. A2). This analysis also allowed me to calculate the position of the p3-pMN boundary. The p3-pMN boundary has shifted dorsally, resulting in a larger p3 domain expressing only Nkx2.2 and a smaller pMN domain (Fig. 5.1D) Ericson et al. (1997). Thus, Pax6 contributes to determining both the position and precision of the p3-pMN boundary.

A potential explanation for the reduced precision in the Pax6$^{-/-}$ is that this is generated by intrinsic noise in production and degradation of proteins in the GRN. We made use of a previous model from the Briscoe lab which was able to correctly position the boundary positions in response to Shh signalling, for both WT and mutant phenotypes including Pax6$^{-/-}$ (Balaskas et al., 2012, Cohen et al., 2014). That model was purely deterministic and therefore could not capture effects of heterogeneity in the *in vivo* sys-

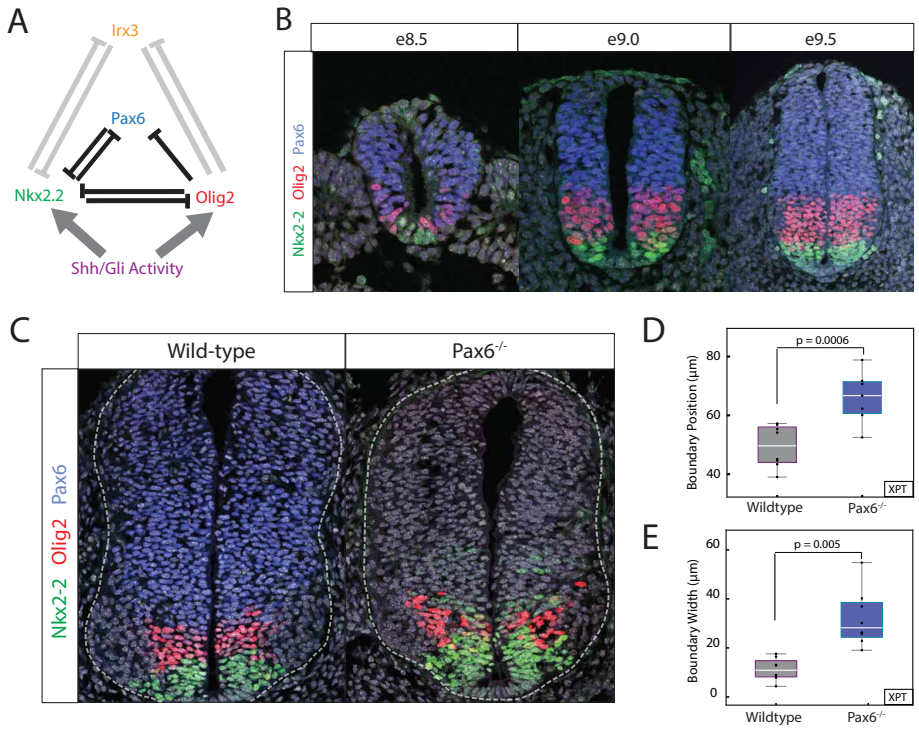


Figure 5.1: Pax6 contributes to boundary precision. Experimental data obtained by Katherine Exelby (**A**) A GRN comprising the transcription factors Pax6, Olig2, Nkx2,2 and Irx3 is responsible for positioning the p3 and pMN domains in the ventral neural tube. (**B**) Immunofluorescence assays of Pax6 (blue), Olig2 (red) and Nkx2.2 (green) indicate dynamic expression in neural progenitors from e8.5 to e9.5. (**C**) Transverse sections of e10.5 WT and Pax6$^{-/-}$ embryos assayed for Olig2, Pax6 and Nkx2.2 expression. (**D**) Position of the pMN-p3 boundary in WT (grey) and Pax6$^{-/-}$ (blue). (Box plots in all figures show upper and lower quartile and mean; $n = 7$ (WT), $n = 8$ (Pax6$^{-/-}$), Mann-Whitney test $p = 0.005$). (**E**) Quantification of pMN-p3 boundary width in WT (grey) and Pax6$^{-/-}$ (blue). Shows boundary is narrower and sharper in WT than Pax6$^{-/-}$ mutants (Mann-Whitney test $p = 0.0006$).

tem. As described in the following section we constructed a SDE model that is based on the ODE system from (Cohen et al., 2014).

5.2. Formulation of stochastic GRN dynamics

In order to investigate heterogeneity of gene expression in the neural tube we make use of stochastic differential equations that describe the GRN and the concentration x_j of each TF j. We start with a thermodynamic-like model as detailed in (Cohen

et al., 2014), which captures the macroscopic behaviour by a system of ODEs; these contain terms for production and decay of each TF. The ODE description corresponds to the limit of a reaction volume Ω that is large enough for the copy numbers Ωx_j of all protein species to be large, allowing fluctuations to be neglected; formally one takes $\Omega \to \infty$. When Ω is finite stochastic effects occur, these can be described by the chemical Langevin equation, a system of SDEs, see e.g. (Gillespie, 2000, Van Kampen, 2007). The drift, *i.e.* the systematic variation with time in the SDEs coincides directly with the deterministic limit. The diffusion (stochastic) term arises from the stochastic nature of the individual protein production and decay reactions; it is a Gaussian white noise (Gillespie, 2000) whose covariance structure is determined by the mean reaction rates. In our case the chemical Langevin equation for the protein levels x_j within the GRN takes the form:

$$\frac{d}{dt}x_j = \sum_{\boldsymbol{n}} p_{(j,\boldsymbol{n})}\alpha_{(j,\boldsymbol{n})} - x_j\beta_j + \Omega^{-1/2}\epsilon_j(t) \tag{5.1a}$$

$$p_{(j,\boldsymbol{n})} = \frac{k_{(j,n)}\prod_i x_i^{n_i}}{\sum_{n'} k_{(j,n')}\prod_i x_i^{n'_i}}$$

$$D_{ij} = \delta_{ij}\left[\sum_{\boldsymbol{n}} p_{(j,\boldsymbol{n})}\alpha_{(j,\boldsymbol{n})} + x_j\beta_j\right] \tag{5.1b}$$

$$\langle \epsilon_i(t)\epsilon_j(t')\rangle = \delta(t-t')D_{ij} \tag{5.1c}$$

The deterministic part of these equations is equivalent to those used in (Cohen et al., 2014). Given that we do not expand our system into mass action reactions in this section, it is convenient to use $p_{(j,\boldsymbol{n})}$ in place of $w_{(j,\boldsymbol{n})}$, and $k_{(j,n)}$ in place of $w_{(j,n)}$. The covariance (5.1b,5.1c) of the zero mean Gaussian white noise $\epsilon_j(t)$ arises from the decay and production of each protein being independent and random, given the concentration of the regulators of the relevant gene. In the equations above, α represents protein production rate and β degradation rate, while the w provide the weights of the

respective DNA conformations $(j, \boldsymbol{n})$ when multiplied by the respective concentration. The conformations are labelled by the protein j being produced and the numbers $\boldsymbol{n} = \{n_i\}$ of TF molecules bound. The δ in (5.1b) and (5.1c) are the Kronecker and Dirac delta respectively. As explained above, Ω is the volume of the system in which all reactions take place.

When looking at the chemical Langevin equation (5.1a), one notices that the rate $\sum_{\boldsymbol{n}} p_{(j,\boldsymbol{n})}\alpha_{(j,\boldsymbol{n})}$ for producing protein j, has a nonlinear dependence on the TF concentrations x_i. One might be concerned that with such a nonlinear dependence, modelling production of protein j as a single reaction is too simplistic. However, (5.1a) can be obtained from a larger system of simple unary and binary mass action reactions, in which the concentration of each DNA conformation is tracked individually. We only sketch this construction here and explain its implications for the stochastic terms in (5.1a); for further details see (Chapter 2). The deterministic part of the time evolution of the DNA concentrations is given as detailed in (2.14) and restated here with the notation from that chapter:

$$\frac{d}{dt}x_{(j,\boldsymbol{n})} = \gamma \sum_p \left(k^{p+}_{(j,\boldsymbol{n}-\boldsymbol{e}_p)} x_{(j,\boldsymbol{n}-\boldsymbol{e}_p)} x_p - k^{p+}_{(j,\boldsymbol{n})} x_{(j,\boldsymbol{n})} x_p + k^{p-}_{(j,\boldsymbol{n}+\boldsymbol{e}_p)} x_{(j,\boldsymbol{n}+\boldsymbol{e}_p)} - k^{p-}_{(j,\boldsymbol{n})} x_{(j,\boldsymbol{n})} \right) \tag{5.2}$$

Here $x_{(j,\boldsymbol{n})} = \tilde{x}_{(j,\boldsymbol{n})}/\gamma'$ tracks the concentration of each DNA conformation and is scaled down by a large factor γ' to account for the low quantity of binding sites in relation to protein numbers. Correspondingly the protein production rate constants $\alpha_{(j,\boldsymbol{n})} = \gamma' \tilde{\alpha}_{(j,\boldsymbol{n})}$ have to be large in order to give an appreciable overall rate of protein production nonetheless.

To derive the correct stochastic equations for the protein species, the large γ-limit of (2.14) is taken: the concentration of each DNA conformation then changes sufficiently quickly that it constantly tracks the instantaneous protein concentrations. For

appropriately chosen binding and unbinding rate constants $k^{p+}_{(j,\boldsymbol{n})}$ and $k^{p-}_{(j,\boldsymbol{n})}$ this leads back to the thermodynamic-like form of the *deterministic* part of the protein dynamics in (5.1a & Chapter 2). As shown in (Thomas et al., 2012a) the existence of fast species (in our case, DNA conformations) can lead to additional terms arising in the *noise* acting on the slow species (protein production), as a consequence of reactions between the slow and fast species. In our case it turns out that these extra noise terms scale with γ'/γ. We then make use of the biological meaning of the terms: $1/\gamma$ represents the timescale of reaction rates for TF binding to DNA and $1/\gamma'$ represents the characteristic time for the process of going from active DNA to producing a protein. We find it biologically reasonable to choose a $1/\gamma$ that is substantially smaller than $1/\gamma'$, given the many biological processes necessary for the production of a fully functional protein. The ratio γ'/γ is hence small, so that the additional noise terms that arise from the general calculation in (Thomas et al., 2012a) become negligible, leaving exactly the noise terms in (5.1c). The intuition is that because protein production is slow compared to binding and unbinding of factors to DNA, noise from the many binding and unbinding events during production averages out; the overall noise then arises only from the stochasticity of the production processes, at the relevant average DNA concentrations. We note that in accordance with this conclusion, explicit calculations show that when γ' is of the order of γ, or larger, then additional noise terms from the stochasticity in DNA concentrations do enter the dynamics of the protein concentrations. Moreover, these additional terms are dependent on the precise choices of binding and unbinding rates, which are only partially constrained by the requirement that the thermodynamic-like deterministic equations (5.1a) are retrieved for large γ (Chapter 2).

5.3. Establishing system parameters

5.3.1. Amount of noise

The stochastic model we are working with is a coarse-grained description that does not explicitly describe the many possible sources of noise within a living cell. These include spatial heterogeneity and effects from the bursty, multi-step nature of protein production, which includes processes such as transcription, translation, post-translational modification, protein folding and protein shuttling (McAdams and Arkin, 1997).

The noise level in our model is set by Ω^{-1}, the inverse reaction volume. This determines the scale of the stochastic fluctuations in protein production and decay, both of which the model represents as single step processes. A larger Ω thus leads to smaller stochastic effects. In equation (5.1a), multiplying Ω by the concentration of a protein species gives the total number of molecules for that protein. In our calculations we measure volumes in units that make typical protein concentrations of order unity, so that Ω can be directly interpreted as a copy number. In accordance with our observations in (Appendix Sec. A.2.4), a value for Ω can be read as a copy number for Pax6, Nkx2.2 and Irx3; the corresponding typical copy numbers for Olig2 are ten times higher (Appendix Sec. A.2.4).

We estimate a lower bound on the noise level Ω^{-1}, *i.e.* the lowest amount of noise that makes sense within our description. It is given by the typical number of proteins of each species in a cell: these numbers determine the minimum amount of noise that must arise from the stochastic nature of protein production and decay. From protein quantifications (Appendix Sec. A.2.4, performed by Lorena Garcia Perez) we obtain $\Omega_{\max} \sim 10,000$ for the protein counts of Nkx2.2 and Pax6 per cell at saturation levels (which in our model correspond to concentrations close to unity). Olig2 has a higher estimated count of ∼100,000 and in accordance a 10 times higher concentration in the model (the

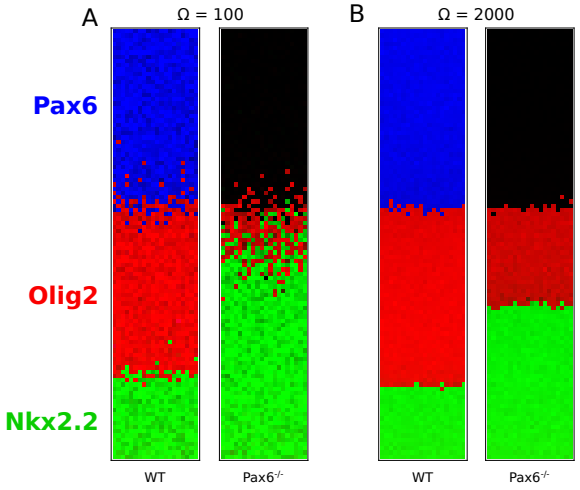


Figure 5.2: Simulations of the WT and Pax6$^{-/-}$ stochastic models for (A) $\Omega = 100$, (B) $\Omega = 2000$. For this range for Ω the simulations recapitulate the observed relationship of boundary sharpness and position in WT and Pax6$^{-/-}$ mutants.

maximum concentration for Olig2 is 10, and 1 for the other TFs), as specified these measurements were obtained by Lorena Garcia Perez. Because of the many neglected sources of additional noise, we expect $1/\Omega_{\max}$ to be a considerable underestimate; indeed, simulations with this noise level show almost deterministic behaviour. However for a slightly increased noise level ($\Omega = 2000$), we find that the relationships between jump-rate differences across WT and mutant phenotypes discussed in the main text hold already (see Fig. 5.2). This means that the WT presents a small amount of heterogeneity (as observed *in vivo*) and the mutants have a more heterogeneous boundary than the WT.

To obtain a lower bound for Ω, we measure the coefficient of variation at steady state for all 3 TF values across embryos, to estimate the total amount of noise in the system (Fig. 5.1A). We then decrease Ω in our numerical simulations until we see coefficients of variation similar to those observed *in vivo*, giving $\Omega_{\min} = 20$. This assumes that *all* observed differences in protein levels arise *solely* from the stochasticity in our model. We reason that there are other sources of noise that make the coefficients of variation higher *in vivo*, such as protein transport within the cell, antibody specificity and measurement

error, so that the amount of noise contributed by the stochasticity in our dynamical model will be smaller than $1/\Omega_{\min} = 1/20$. On that basis we find a reasonable smallest value of Ω of ~ 100. The value we use for all results throughout this study is $\Omega = 250$, which is within the broad bounds of $\Omega_{\min} = 20$ and $\Omega_{\max} = 20,000$. Importantly, the results we observe remain qualitatively unchanged across the entire range of Ω that we assess as reasonable, $100 \leq \Omega \leq 2000$ (Fig. 5.2).

5.3.2. Initial conditions and consistency across mutants

Following the inclusion of the noise term and its parametrisation, we explored the effect of the initial conditions for the TFs (i.e. their initial expression levels x_j). The aim was to find a consistent set of initial conditions that sustain the boundary positions but also recapitulate the boundary sharpness of each mutant. The initial conditions that satisfied these conditions were identified in a systematic scan as $x_{\mathrm{Pax6}} = 0.1$, $x_{\mathrm{Olig2}} = 0$, $x_{\mathrm{Nkx2.2}} = 0$, $x_{\mathrm{Irx3}} = 0.1$. The p3-pMN boundaries in WT, Irx3$^{-/-}$, Nkx2.2$^{-/-}$ and Olig2$^{-/-}$ simulations remained sharp as is the case *in vivo* (Fig. 5.3). Only the loss of Pax6 resulted in decreased boundary sharpness, removing either Nkx2.2 or Olig2 had the already known effect on the positions of boundaries between domains but did not affect boundary precision, consistent with known literature This agrees with experimental observations (Balaskas et al., 2012, Briscoe et al., 1999, Novitch et al., 2001). Boundary positions remained consistent with *in vivo* observations as was the case in the original deterministic model (Fig. 5.3 & Fig. 5.4A-C) & (Cohen et al., 2014). We conclude from this that by including intrinsic noise in the GRN dynamics we can accurately predict the known changes in precision of gene expression while retaining correct boundary positions.

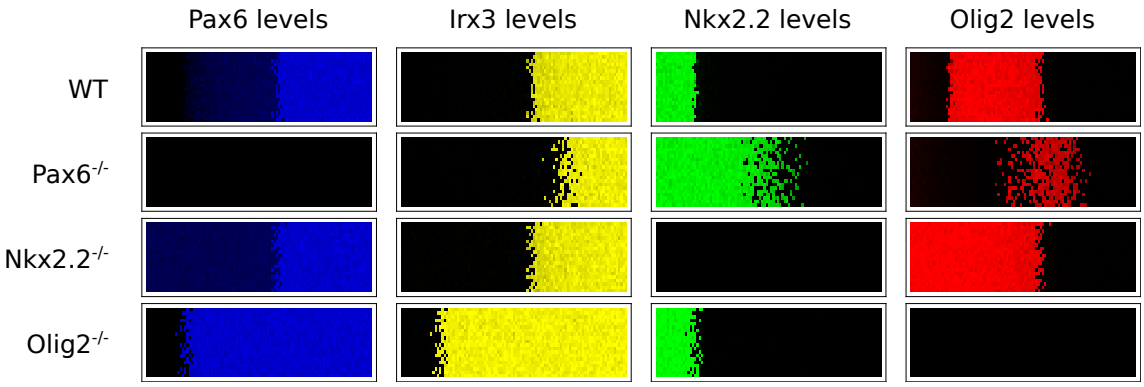


Figure 5.3: Patterning phenotypes produced by stochastic simulations for WT and mutants. Predicted expression patterns for the four TFs in the indicated genotypes are qualitatively similar to those in (Cohen et al., 2014). Ventral to the left and dorsal to the right. Although boundary positions change, boundary precision is largely unaffected except for Pax6$^{-/-}$, consistent with *in vivo* experimental observations.

5.4. Boundary width is determined by changes in transition times

We seek to understand how intrinsic noise leads to imprecise boundaries. The model did not predict changes in the magnitude of fluctuations in gene expression between the WT and Pax6$^{-/-}$ (Section A.4), so we went on to test this experimentally. We obtained the coefficient of variation (CV) of Olig2 as it is the only comparable expressing gene in the pMN region and found WT (CV: 0.42) and Pax6$^{-/-}$ (CV: 0.44), indicating no significant differences (Mann-Whitney p=0.422). This suggests that the change in boundary precision is not caused by a change in the size of fluctuations but by the dynamical landscape generated by the network. We highlight that the region of heterogeneity is generated in a zone of bistability, with the domains of high Nkx2.2 (p3) and high Olig2/Pax6 (pMN) (Fig. 5.4D). In the absence of noise, the transition between the two steady states is determined solely by the level of Shh signalling, and the system remains in the pMN state until the level of signalling increases above the bistable region. However, in the presence of intrinsic noise, fluctuations in gene expression can result in spontaneous transitions between pMN and p3 identity within the bistable region Perez-Carrasco et al. (2016). Below a threshold of Shh signalling, the rate of transitions is very low and cells remain

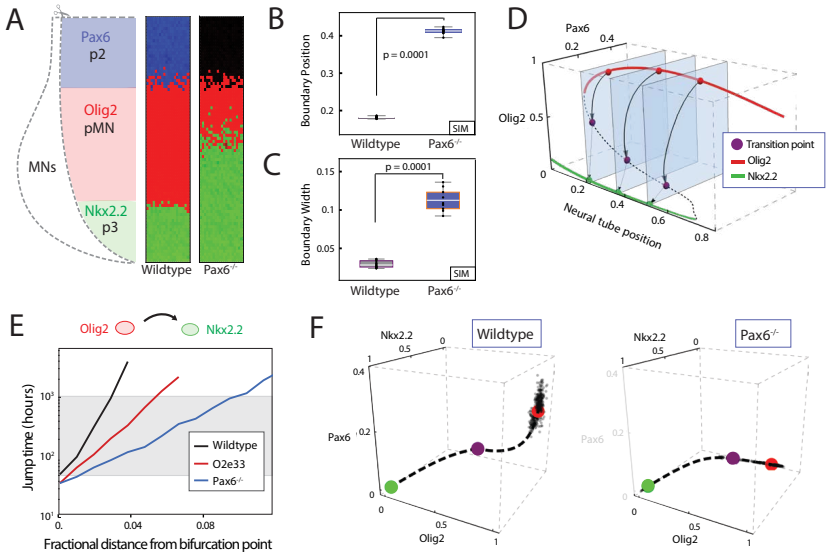


Figure 5.4: Transition times determine boundary precision. (**A**) Simulations of the stochastic dynamics of the GRN account for the experimentally observed dorsal expansion of the p3 domain and loss of boundary precision in Pax6$^{-/-}$ (right) compared to WT (middle). Diagram (left) illustrates the pattern of gene expression in WT neural tube with progenitor domains (p2, pMN, p3) and differentiated neuron populations (motor neurons, MNs). (**B,C**) Quantification of boundary position and width from simulations: the boundary in Pax6$^{-/-}$ (blue) is wider (less sharp) than WT (grey). Width is given as fraction of total neural tube size. $n = 10$ (WT), $n = 10$ (Pax6$^{-/-}$), Mann-Whitney test $p = 0.0001$ for position, $p = 0.0001$ for boundary width. (**D**) A 3D bifurcation diagram of the mathematical model illustrates a region of bistablity for pMN (red; expressing Olig2 and Pax6) and p3 (green; expressing Nkx2.2) separated by a transition point (unstable fixed point of dynamics, purple). The noise driven transition pathway from pMN to p3 is indicated by black arrows (solid: pMN to transition point, dashed: transition point to p3). (**E**) Fate jump times calculated from simulations: average time for noise driven transitions from pMN to p3 in WT (black), Pax6$^{-/-}$ (blue) and O2e33 mutants (red). Relative position refers to distance from the bifurcation point. Grey shading indicates the time regime where transitions can occur on relevant developmental timescales. In WT, jump times change more rapidly as a function of position, leading to a narrower boundary than in Pax6$^{-/-}$. (**F**) Gene expression space view of the transition path from pMN (red point) to p3 (green point) steady state via the transition point (purple point). Simulated trajectory (dots) shows stochastic fluctuations from the pMN steady state remain close to the most likely transition path (dotted line) and are not oriented directly towards the transition point in WT (left). Axes show relative expression levels. In Pax6$^{-/-}$ (right) the loss of the Pax6 dimension means that fluctuations are oriented towards the transition point. Hence, similar noise levels will result in more fluctuation-induced jumps. Data are shown for neural tube position at 0.1 fraction of total neural tube length dorsal to the bifurcation point and are representative of the behaviour within the bistable region.

in the pMN state. Conversely, above a certain level of Shh signalling, transitions from the pMN to the p3 steady state take place so rapidly that essentially all cells undergo this transition and become Nkx2.2 positive. In between these two regimes, a region of heterogeneity is observed in which there is an intermediate probability for each cell of a spontaneous transition on a developmentally relevant timescale ($\leq$50 hours). The size of this region of heterogeneity depends on how the probability of a noise induced transition changes in response to alterations in the level of Shh signalling. We calculated the characteristic time it would take for transitions between the pMN and p3 states at different dorsal-ventral positions of the neural tube. We termed this "fate jump time". For WT, fate jump time changes rapidly in response to Shh signalling, implying that there is only a limited region where the effective probability of transitions is not 0 or 1 (Fig. 5.4E; black line). By contrast, the larger region of heterogeneity observed in the Pax6$^{-/-}$ mutant is due to the weaker dependence of fate jump time on levels of Shh signalling (Fig. 5.4E; blue line). There is a larger range of Shh levels for which noise driven transitions are possible and therefore a larger boundary region where cells in both Nkx2.2 or Olig2/Pax6 states exist.

5.5. Elucidating sensitivity to signal

This prompted us to investigate why there are substantial differences in the rate at which the transition time changes with respect to position. To understand thi we analysed the dynamics of transition between the two steady state and how they passed through the corresponding saddle point in phase space (Fig. 5.4F; purple point). Simulations of the SDE model indicated that intrinsic fluctuations around the pMN state are initially directed away from the transition point in WT. By contrast, in the Pax6 mutant fluctuations are oriented directly towards the transition point.

To characterise this rigorously, we calculated the minimum action path (MAP) between

the pMN and p3 steady states: this predicts the most likely gene expression trajectory that a stochastic transition resulting from a small fluctuation in gene expression will take, thereby providing a portrait of the dynamical landscape that leads to a noise induced transition (Fig. 5.4F, this is expanded upon in the following subsection) Bunin et al. (2012), Kleinert (2009), Perez-Carrasco et al. (2016).

5.5.1. Minimum action path

Much of the theoretical analysis in this chapter concentrates on the stochastic transitions between fixed points of the deterministic GRN dynamics, which are long-lived metastable states of the stochastic dynamics. The MAP as specified above is the most likely path the system takes in such a transition (for large enough values of Ω), from a steady state to a transition point (which is the saddle point of the dynamical system) and then onwards to a new steady state. The second piece of the path always follows the deterministic dynamics and has a negligible effect on the transition times, so we focus on the first part of the path.

The negative log probability for any path is proportional to what is called the action, which for our Langevin dynamics is of so-called Onsager-Machlup form (Kleinert, 2009). The action is an integral over time of the Lagrangian, which in turn depends only on the current state (vector of concentrations) and velocity of the system. The time integral can be discretised and the action then minimised as described in e.g. (Bunin et al., 2012). We analyse the resulting MAP in gene expression space in order to understand how its shape affects the jump times between steady states and thus eventually the boundary width.

The typical time the system takes to reach any point on the MAP scales exponentially with the action up to that point, hence this quantity can be interpreted as an effective energy, within the analogy of a particle making a transition from one local minimum in

an energy landscape across a barrier to another minimum. Fig. 5.5 shows an alternative representation that gives further insight: we plot the derivative of the action along the path, which is the effective force pushing the system back towards the initial steady state.

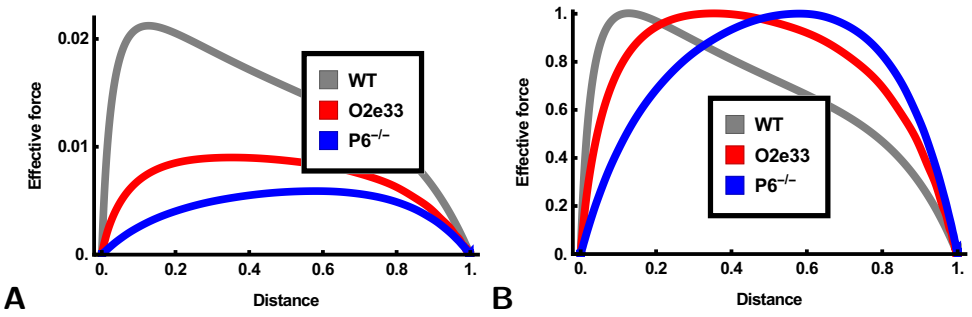


Figure 5.5: (**A**-**B**) Unnormalised and normalised space derivative of the action 5.9**D**(i) along the MAP, plotted along the length of the path. This reflects the effective force driving the system back towards its initial steady state. In the WT system (gray) the force is highest near the beginning of the path, leading to a noticeably skewed plot, while the O2e33 (red) and Pax6$^{-/-}$ (blue) more nearly symmetric force profiles. The high initial force in WT responsible for the large typical jump times in the system, and is related to the significant curvature of the MAP away from the straight line between initial steady state and transition point (Fig. 5.9A-B)

Consistent with the SDE simulations, in WT, the MAP from the pMN (Olig2/Pax6) to p3 steady state does not follow the shortest route leading to the transition point, instead the levels of Pax6 drop rapidly and pitch away from the transition point, resulting in a curvature of the gene expression path between steady states (Fig. 5.4F). By contrast, in the absence of Pax6, the MAP is directly oriented towards the transition point (Fig. 5.4F). Taken together, the analysis suggests that the GRN affects the precision of a domain boundary by determining the dynamical landscape, without changing the level of noise in overall gene expression.

5.6. Perturbing interactions: O2e33 enhancer mutant

After this conclusion regarding the noise levels in the Pax6$^{-/-}$ we realised it would be good to find less dramatic examples that don't involve removing a dimension entirely. These systems should also be testable experimentally to confirm any theoretical observations. This led to a search of other systems with reduced precision between domains while retaining roughly the same level of noise in overall gene expression. All knock-outs of genes had been performed previously, so we looked to perturbations that affect cis-regulatory elements that control interactions between TFs.

Katherine Exelby performed a knock-out of a well characterised Olig2 enhancer, we summarize here the experimental setup and results, as described in (Exelby et al., 2019):

> "Several predicted regulatory regions are located in the vicinity of Olig2; these include a prominent candidate region 33kb upstream of the Olig2 gene Oosterveen et al. (2012), Peterson et al. (2012), which we termed O2e33. This region binds (i) the repressor Nkx2.2, (ii) Sox2, which activates Olig2, and (iii) the Gli proteins, the transcriptional effectors of the Shh pathway (Fig. 5.6A) Kutejova et al. (2016), Nishi et al. (2015), Oosterveen et al. (2012), Peterson et al. (2012). To test the role of O2e33 in the network, its function was first analysed *in vitro* in neural progenitors differentiated from mouse embryonic stem (ES) cells Gouti et al. (2014). Using CRISPR/Cas9, the ∼1kb enhancer region was excised in ES cells harbouring an Olig2 fluorescent reporter Sagner et al. (2018). Unlike WT cells, which express high levels of Olig2 at Day 5 of differentiation Gouti et al. (2014), Sagner et al. (2018), cells lacking the O2e33 enhancer had a marked reduction in levels of Olig2. By Day 6, Olig2 expression had increased in O2e33 mutant cells, but the percentage of cells and the level of expression never reached that of

WT (Fig. 5.6B,C)."

5.6.1. Finding parameters that explain O2e33 phenotype

To find parameter sets that describe the behaviour of the O2e33 enhancer mutation, we first identified those parameters that are related directly to the deletion of the respective enhancer. Analysis of the sequence of the enhancer together with CHIP-seq and ATAC-seq (Kutejova et al., 2016, Metzis et al., 2018, Oosterveen et al., 2012, Peterson et al., 2012) suggested that Gli proteins, Nkx2.2, Irx3, and Sox2 all have a direct effect on this enhancer (Fig. 5.6A).

Aided by this biological knowledge, we accounted for variations in the parameters that specify Nkx2.2 binding, Irx3 binding, Gli binding and basal production (corresponding to Sox2 binding). We systematically explored how reducing the parameters for each of these interactions, to a fraction f of their original value, could explain the observed phenotype. We used a uniform distribution to perform this search and represent the respective parameter reductions directly in terms of the ratio f between new and original (WT) parameter values.

We first identified parameter sets that could replicate the observed *in vitro* delay in the onset of Olig2 expression in the mutant, leading to a reduced parameter space (Fig. 5.7). The delay in Olig2 activation was determined for networks positioned a fraction 0.3 along the neural tube, and we retained those networks that took twice the amount of time to express Olig2 than in the WT.

We next investigated what further phenotypical behaviour the retained parameter sets predict, focussing on the domain size and boundary precision generated in response to a graded Shh signal. We found that 68% of the parameter combinations reduced boundary precision, 80% reduced the size of the pMN domain, with 83% presenting one

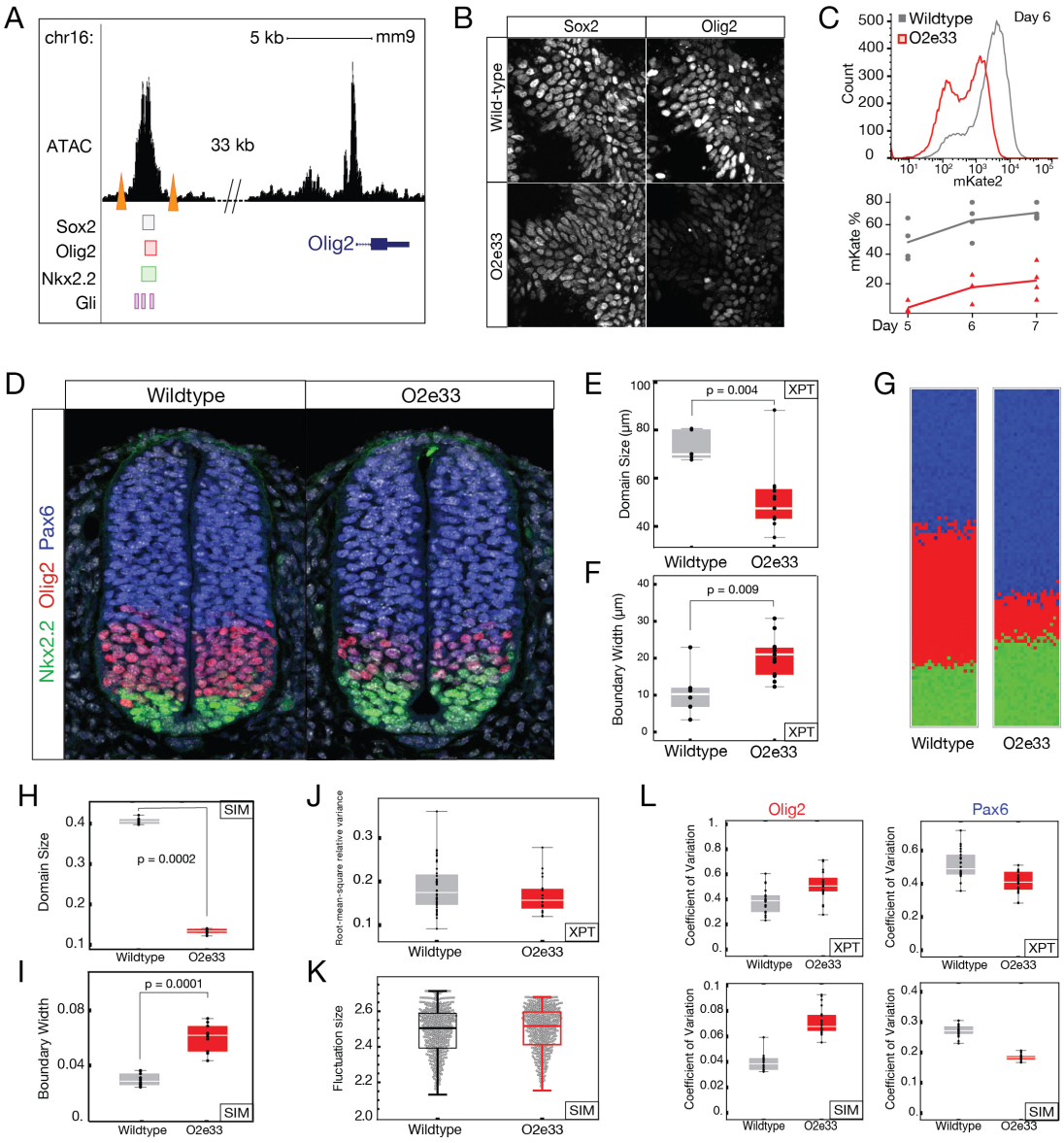

A
chr16:
5 kb
mm9
ATAC
33 kb
Sox2
Olig2
Nkx2.2
Gli
Olig2
B
Sox2
Olig2
Wild-type
O2e33
C
Wildtype
O2e33
Day 6
Count
mKate2
mKate %
Day 5
6
7
D
Wildtype
O2e33
Nkx2.2 Olig2 Pax6
E
p = 0.004
XPT
Domain Size (µm)
Wildtype
O2e33
F
p = 0.009
XPT
Boundary Width (µm)
Wildtype
O2e33
G
Wildtype
O2e33
H
SIM
Domain Size
p = 0.0002
Wildtype
O2e33
I
p = 0.0001
Boundary Width
SIM
Wildtype
O2e33
J
Root-mean-square relative variance
XPT
Wildtype
O2e33
K
Fluctuation size
SIM
Wildtype
O2e33
L
Olig2
Pax6
Coefficient of Variation
XPT
SIM
Wildtype
O2e33

Figure 5.6: An Olig2 enhancer affects precision of the pMN-p3 boundary. (**A**) Chromatin accessibility (ATAC-seq) and predicted TF binding locations based on ChIP-seq data highlight a putative regulatory element 33kb 5' of Olig2. CRISPR target sites (orange triangles) indicate the region excised to create the O2e33 deletion Kutejova et al. (2016), Metzis et al. (2018), Oosterveen et al. (2012), Peterson et al. (2012). (**B**) Immunofluorescence for Sox2 (expressed in all neural progenitors) and Olig2 at day 6 in neural progenitors differentiated from WT and O2e33 ES cells exposed to 500nM SAG. Reduced levels of Olig2 are seen in O2e33 mutant cells. (**C**) Flow cytometry (top) for mKate2 flourescence in Olig2-T2A-mKate2 ES cell derived neural progenitors at day 6 of differentiation, exposed to 500nM SAG, indicating the decreased proportion of O2e33 cells (red) expressing Olig2 and the lower levels of Olig2 expression compared to WT cells (grey). Lower percentage of mKate2 flourescent cells between days 5-7 of differentiation in WT and O2e33 cells (bottom). (**D**) Immunofluorescence for Olig2, Pax6 and Nkx2.2 in transverse sections of e9.5 neural tube from WT and O2e33 embryos revealing a change in the position of the pMN domain (red; Olig2 expressing) and the sharpness of its boundary with p3 progenitors (green; Nkx2.2 expressed). (**E, F**) Quantification of domain size and boundary width in WT (grey) and O2e33 mutants (red). The pMN domain in WT is larger than in O2e33 embryos at e9.5 (Mann-Whitney test $p = 0.004$). The p3-pMN boundary is wider, i.e. less sharp, in O2e33 mutants compared to WT ($n = 6$ (WT), $n = 12$ (O2e33), Mann-Whitney test $p = 0.009$). (**G**) Stochastic simulations of the O2e33 model recapitulate *in vivo* observations of a narrower pMN domain and decreased precision of the p3-pMN boundary. (**H, I**) Quantification of boundary width (I) and position (H) from simulations demonstrates loss of precision and change in pMN domain size in O2e33 simulations compared to WT, consistent with *in vivo* observations (Box plot shows upper and lower quartile and mean; n=10 (WT); n=10 (O2e33), Mann-Whitney test p=0.0001 for both). (**J**) Quantification of total variance in gene expression per embryo (Olig2 and Pax6) within the pMN domain for WT (grey) and O2e33 embryos (red). Relative root-mean-square variance of WT and O2e33 embryos captures total noise of the system (spread of points). A Mann-Whitney test indicates no significant change in total noise levels between genotypes ($p > 0.05$). (**K**) Measurements of noise at all positions in the pMN domain performed *in silico*. The magnitude of the relative noise variance at all possible configurations of the system indicates similar levels of noise in WT and O2e33, each grey point is an individual configuration (Section A.4, Mann-Whitney test $p > 0.05$). (**L**) Coefficient of variation values were calculated for Olig2 (left) and Pax6 (right) expression in both WT (grey) and O2e33 (red) from experimental data (top) and *in silico* simulations (bottom). This showed an increase in the variation of Olig2 in the O2e33 compared to the WT as well as a decrease in variation of Pax6.

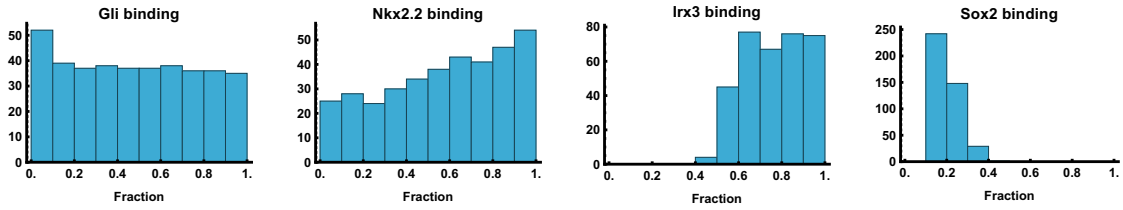


Figure 5.7: Distribution of parameter changes to mimic *in vitro* O2e33 mutant. To recapitulate the O2e33 dynamics *in vitro*, model parameters were systematically explored to identify changes that could account for the delay in onset of Olig2 expression. The graphs show the distributions of reduction factors f (x-axis) relative to WT parameter values, across parameter sets that recapitulate the delay. The (y-axis) shows number of parameter combinations that recapitulate the phenotype. The results show that what is needed to generate a delayed induction of Olig2 is a substantial reduction in Sox2 input while maintaining input of Irx3.

or other of the phenotypes. Here, the pMN domain size was calculated with respect to the Shh gradient and we considered it reduced if it was below 70% of the WT size. For determining boundary sharpness, we regarded as imprecise those systems that had a boundary width at least twice the size of the WT; this width is calculated using the SDE system with the thresholds described in Fig. 5.4E. The fact that a majority of the parameter sets identified affected domain size and boundary precision encouraged us to generate the mouse lines.

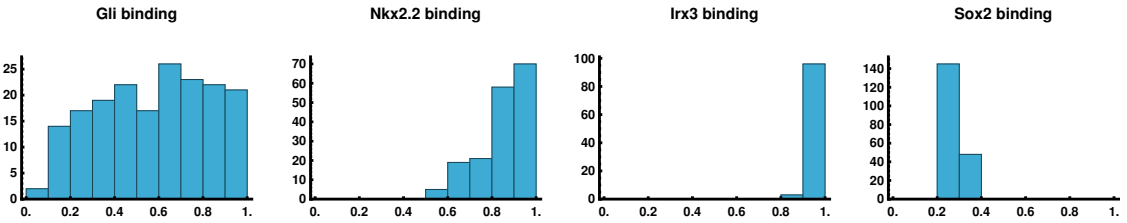


Figure 5.8: Distribution of parameter changes to mimic *in vivo* O2e33 mutant. Equivalent histograms to Fig. 5.7 with the additional constraints from *in vivo* observations: ventral shift of pMN-p2 boundary *and* broad p3-pMN boundary. The main results are: maintaining WT levels of Irx3 input; substantial reduction in Sox2 input, some reduction in Gli input but with a broad distribution, and a mild reduction in Nkx2.2 input.

In the experimentally generated mouse lines (App. A.2) two phenotypes were notable

in addition to the delay in onset of Olig2, as expected from the initial parameter screen: strikingly, a loss of precision at the p3-pMN boundary and a ventral shift of the pMN-p2 boundary (Fig. 5.6D). The decrease in boundary precision is observed even though there is sustained expression of Olig2 and Pax6 in pMN cells and Nkx2.2 in p3 cells. This strongly suggests that secondary correction mechanisms are not playing a major role in setting the precision of the boundary between these two domains.

We made use of these two additional observations to constrain our parameter space further, thus leading to the parameter distributions shown in Fig. 5.8. We quantified sharpness as explained above and confirmed the decrease in boundary precision as well as the smaller pMN domain (Fig. 5.6E,F). The targets set for the boundary position were extracted from *in vivo* data, and were set as: pMN-p3 boundary position to be between [0.17 0.25] (as the WT boundary position is at 0.17 and the in vivo data show a small dorsal shift in the mutant) and p2-pMN position to be lower or equal to 0.5 (WT boundary is at 0.55, this means a reduction of the domain size of at least 15% with respect to WT) but higher than the pMN-p3 boundary position, such that the pMN domain does not disappear. From the parameter sets that met both criteria, we finally took a representative point as our model for the O2e33 mutant; as expected this replicates the observed experimental phenotypes (Fig. 5.6G).

This produced simulations in which the loss of boundary precision in the O2e33 is not as severe as the Pax6$^{-/-}$ phenotype, consistent with the experimental data (Fig. 5.6G). Boundary width and pMN domain size from simulations were consistent with the *in vivo* analysis (Fig. 5.6H & I). Calculating the pMN to p3 fate jump times revealed that for the O2e33$^{-/-}$, fate jump times changed more slowly than for WT (Fig. 5.4E), in line with the decreased boundary precision of O2e33$^{-/-}$. Analysis *in vivo* of the magnitude of the combined fluctuations in Pax6 and Olig2 indicated that it was similar in WT and O2e33$^{-/-}$ (Fig. 5.6J; App. A.3). Consistent with this, the combined magnitude of

fluctuations of Pax6 and Olig2 in simulations were similar in WT and O2e33 mutants, suggesting the decreased precision was not the result of an increase in the overall size of the noise driven fluctuations (Fig. A6)(Fig. 5.6K). However, the simulations predicted that variability in Olig2 should increase while the variability of Pax6 should decrease. In agreement with this prediction, the coefficients of variation of Olig2 and Pax6 gene expression between WT and O2e33 *in vivo* were increased and decreased, respectively (Fig. 5.6L).

5.7. Shape of the transition path affects boundary

To investigate why this led to a decrease in boundary precision, we analysed the MAP of the O2e33$^{-/-}$ at a fixed neural tube position. The model indicated that the transition path from pMN to p3 curved away from the shortest path to a lesser extent than for the WT; stochastic simulations further confirm this behaviour (Fig. 5.9A & B). Thus, in the absence of the O2e33 enhancer, stochastic fluctuations around the Olig2/Pax6 steady state tended to take the system closer to the transition point than similar magnitude fluctuations in WT, making a noise driven switch in fate more likely in the mutant. Nevertheless, the curvature in the path in O2e33 was greater than in the Pax6$^{-/-}$, providing an explanation for the greater imprecision in Pax6$^{-/-}$ compared to O2e33 mutant (Fig. 5.9B cf. Fig. 5.4E & F). We calculated the action along the path for each genotype de la Cruz et al. (2018)(Fig. 5.9C & Section 5.5.1). This represents the effective energy required to reach a point along the MAP and is a measure of the extent of the barrier that has to be overcome for a fate transition. Consistent with the results of the simulations, the effective energy necessary for a noise induced transition was greatest for WT, less for O2e33, and lowest for the Pax6 mutant. Moreover, the analysis indicated that the initial part of the trajectory presented a more significant barrier to noise induced transitions in the WT than O2e33 and Pax6 mutants (Fig. 5.2A), corresponding

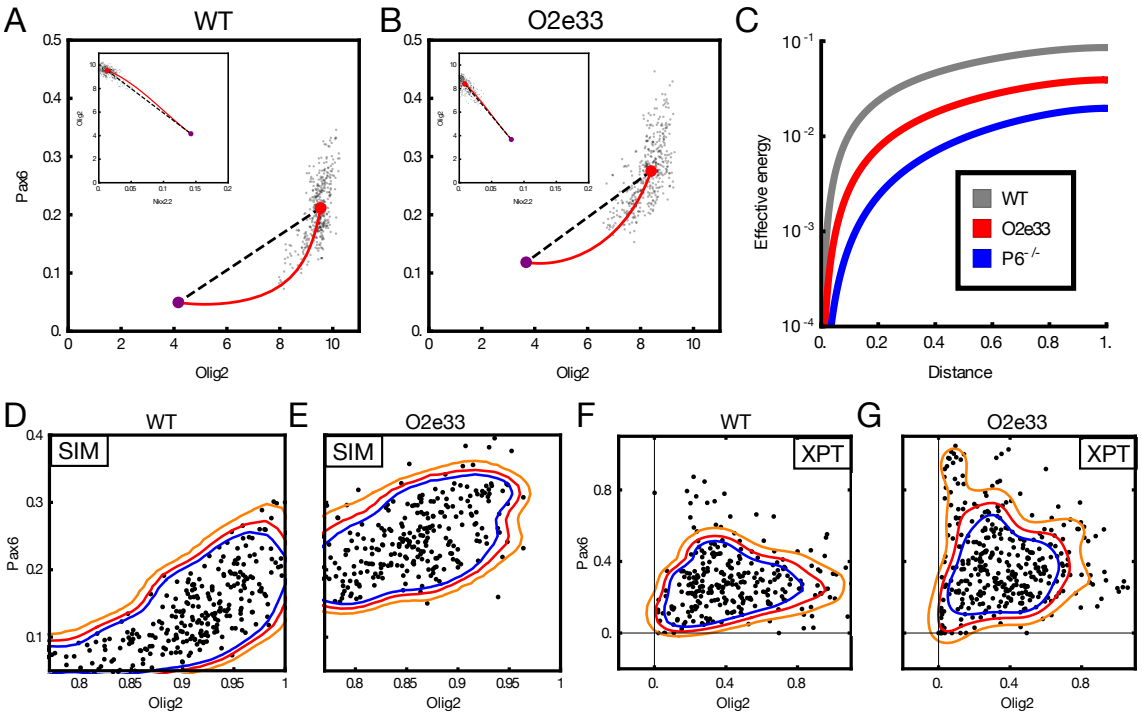


Figure 5.9: The Olig2 enhancer affects the configuration of gene expression fluctuations. (**A**-**B**) Projection into Olig2-Pax6 gene expression space of the minimum action path (red) predicted from the model and simulated trajectory (dots) in WT (A) and O2e33 (B). Rather than follow a direct trajectory (dotted line) from the pMN steady state (red dot) to the transition point (purple dot), gene expression initially curves away from it. However curvature is less marked in O2e33 simulations than WT. Insets show projection onto Nkx2.2-Olig2 axes indicating that in this plane the paths lead almost directly to the transition point. (**C**) Effective energy barrier (cumulative action) for noise-induced transitions, plotted along the transition path (normalised to unit length) at the same neural tube positions as in Fig. 5.4F. WT (grey) has a higher barrier than O2e33 (red), leading to longer jump times; O2e33 in turn has a higher barrier than Pax6$^{-/-}$ (blue). (**D**-**E**) Simulated Pax6 and Olig2 expression levels (black dots) for WT and O2e33 in regions proximal to the p3-pMN boundary. The simulations predict an increase in Pax6 levels and a decrease in Olig2 levels for pMN cells near the boundary in O2e33 mutants. (**F**-**G**) Corresponding to simulation results, a shift to higher levels of Pax6 and reduced levels of Olig2 is observed in cells from O2e33 mutants *in vivo* compared to controls. Axes show fluorescence intensity (arbitary units), see Supplemental Section A.3

to the relative divergences of their transition trajectories from the shortest route to the transition point.

The model predicted that the deletion of O2e33 alters the relative expression levels of Olig2 and Pax6 in individual cells, resulting in cells close to the pMN-p3 boundary

expressing higher levels of Pax6 and lower levels of Olig2 in O2e33 mutants than in WT (Fig. 5.9D, E). We used single cell immunofluorescence quantifications to compare cells in the boundary region of WT and O2e33 embryos (Fig. 5.9F, G & Supplemental Section A.3). Consistent with the predictions, this revealed higher levels of Pax6 and lower levels of Olig2 in O2e33 mutants compared to WT. Thus the experimental evidence supports the idea that the strength of regulatory interactions encoded in the GRN contributes to the precision of domain boundaries by configuring the dynamics of stochastic fluctuations to reduce the probability of a noise driven change in cell identity.

In this chapter we have provided detailed understanding of the mechanism through which cells make collective decisions that lead to a precisely patterned tissue. In conjunction with experimental results this shows that the increase precision is not a result of reducing noise levels of individual genes. Furthermore, the mechanism we have described does not rely on cell-cell communication. We emphasise how it would not be possible to understand this behaviour as the result of a single gene but that it needs to be studied as a multidimensional dynamical system. Gradients of morphogen protein are prevalent across biology, raising the possibility that the mechanism described in this chapter may be a general property in biological networks, this is further discussed in the following chapter

6

Generalising the emergence of boundary sharpness

The finding that the dynamics produced by the regulatory interactions between Pax6-Olig2-Nkx2.2 influences the pMN-p3 boundary precision prompted us to ask whether this is is the only way in which precision can be enhanced by the GRN or whether other mechanisms can increase the fidelity of gene expression boundaries. In this chapter we performed a computational screen to identify three node networks capable of generating a sharp boundary in response to a graded input (Fig. 6.2A).

6.1. Defining a functional form

To perform a parameter screen we explored three node networks with all possible interactions between the nodes (Fig. 5.1A), as this has provided useful insights in other systems (Cotterell and Sharpe, 2010, Leon et al., 2016, Ma et al., 2009). For the purpose of exploring different dynamics, we enumerated the different possible transcriptional/occupancy states of the promoter to model the production rates of a given protein. These rates depend on polymerase availability, signal input (morphogen) and regulating transcription factors, with concentrations x_{p}, x_{in} and x_i respectively. The transcription factors i can be activating ($i \in \mathcal{P}$) or repressing ($i \in \mathcal{N}$), with $\mathcal{P}$ and $\mathcal{N}$ denoting the sets of activating and repressing transcription factors, respectively. While in the previous model, in its most general form (5.1a), different protein production rates can be used for different DNA conformations, in the neural tube network we used the same the production rate for all protein-producing input conformations (see Appendix. 5.6). We adopt the same approach here and set the production rate to unity in appropriate units of time; thus the model is specified only by the binding affinities of the various DNA conformations. Without loss of generality we fixed the affinity (and hence the weight) of the unbound conformation to 1 as explained in (Sherman and Cohen, 2012). We assign the weights of conformations with only one bound molecule as $k_{\mathrm{p}}x_{\mathrm{p}}$, $k_{\mathrm{in}}x_{\mathrm{in}}$ and $k_i x_i$. In accordance with our previous model (5.1a), we set the following constraints:

- All conformations with polymerase and without any repressor $i \in \mathcal{N}$ produce protein; it does not matter whether signal or any activator $i \in \mathcal{P}$ are bound.
- Conformations that have one or more repressor $i \in \mathcal{N}$ bound together with either signal, polymerase or any activator $\mathcal{P}$ are excluded, based on the assumption that these molecules compete for the same binding site
- Binding of signal or any activator $\mathcal{P}$ enhances binding of polymerase

- No other cooperativity effects are present

6.2. Expressions for conformation states

The only states that can produce protein are those with polymerase bound. For brevity we follow the convention in Appendix 5.6 and abbreviate

$$w_{\mathrm{p}} = k_{\mathrm{p}} x_{\mathrm{p}} \tag{6.1}$$

in the following, taking polymerase levels as constant throughout our dynamics. As specified above, the only states that can bind polymerase are those that have no repressors bound. We assume no cooperativity between signal x_{in} and activators $x_i, i \in \mathcal{P}$, hence the total weight of states that can potentially bind polymerase (assuming two binding sites per activator $i \in \mathcal{P}$ but only one for the signal) is:

$$(1 + k_{\mathrm{in}} x_{\mathrm{in}}) \prod_{i \in \mathcal{P}} (1 + k_i x_i)^2 \tag{6.2}$$

Given that repressors $\mathcal{N}$ can only bind by themselves, and that there is no other cooperativity between the inputs, the total weight for conformations with at least one repressor $\mathcal{N}$ bound while assuming two binding sites per repressor $i \in \mathcal{N}$ is:

$$-1 + \prod_{i \in \mathcal{N}} (1 + k_i x_i)^2 \tag{6.3}$$

In accordance with biological intuition, polymerase is recruited by activators $\mathcal{P}$ or signal. The simplest way to implement this is to increase the weight of conformations having both polymerase and at least one activator $i \in \mathcal{P}$ or signal by a cooperativity factor c,

giving a total weight of:

$$c\, w_{\mathrm{p}}[-1 + (1 + k_{\mathrm{in}} x_{\mathrm{in}}) \prod_{i \in \mathcal{P}} (1 + k_i x_i)^2] \tag{6.4}$$

Finally, the weight for the unbound (empty) conformation is taken as 1, as explained above, and for the conformation with one polymerase bound it is w_{p} as defined in (6.1). The total weight, i.e. the denominator of the protein production rate, is then

$$w_{\mathrm{p}} + c\, w_{\mathrm{p}}[-1 + (1 + k_{\mathrm{in}} x_{\mathrm{in}}) \prod_{i \in \mathcal{P}} (1 + k_i x_i)^2] + (1 + k_{\mathrm{in}} x_{\mathrm{in}}) \prod_{i \in \mathcal{P}} (1 + k_i x_i)^2 - 1 + \prod_{i \in \mathcal{N}} (1 + k_i x_i)^2 \tag{6.5}$$

while the numerator is the total weight of conformations *with* polymerase, either on its own (6.1) or together with activators or signal (6.4), giving overall for the production rate (which with protein production set to unity is also the probability of being in a DNA conformation that produces protein)

$$\frac{w_{\mathrm{p}} + c\, w_{\mathrm{p}}[-1 + (1 + k_{\mathrm{in}} x_{\mathrm{in}})\phi]}{w_{\mathrm{p}} + c\, w_{\mathrm{p}}[-1 + (1 + k_{\mathrm{in}} x_{\mathrm{in}})\phi] + (1 + k_{\mathrm{in}} x_{\mathrm{in}})\phi + \psi - 1} \tag{6.6}$$

with the abbreviations

$$\phi = \prod_{i \in \mathcal{P}} (1 + k_i x_i)^2, \qquad \psi = \prod_{i \in \mathcal{N}} (1 + k_i x_i)^2 \tag{6.7}$$

6.3. General strong cooperativity limit

It will be convenient in the following to write the effective affinities of signal and activating TFs in combination with polymerase in a form that includes the cooperativity effect from the factor c, i.e. in terms of $\tilde{k}_{\mathrm{in}} = c\, k_{\mathrm{in}}$ and $\tilde{k}_i = c\, k_i$ for $i \in \mathcal{P}$. The protein

production rate is then expressed as

$$\frac{w_\mathrm{p} + c\, w_\mathrm{p}[-1 + (1 + \tilde{k}_\mathrm{in} x_\mathrm{in}/c)\phi]}{w_\mathrm{p} + c\, w_\mathrm{p}[-1 + (1 + \tilde{k}_\mathrm{in} x_\mathrm{in}/c)\phi] + (1 + \tilde{k}_\mathrm{in} x_\mathrm{in}/c)\phi + \psi - 1} \tag{6.8}$$

with now

$$\phi = \prod_{i \in \mathcal{P}} (1 + \tilde{k}_i x_i/c)^2 \tag{6.9}$$

We can now compare with the analogous expression (A.1.1) in the neural tube network. There all interactions are repressive so that $\mathcal{P}$ is the empty set and hence $\phi = 1$, which simplifies (6.8) to

$$\frac{w_\mathrm{p}(1 + \tilde{k}_\mathrm{in} x_\mathrm{in})}{w_\mathrm{p}(1 + \tilde{k}_\mathrm{in} x_\mathrm{in}) + \tilde{k}_\mathrm{in} x_\mathrm{in}/c + \psi} \tag{6.10}$$

This agrees with (A.1.1) except for the middle term in the denominator, which represents the weight of DNA conformations with only signal but no polymerase bound. Its absence in the neural tube network formally corresponds to the strong cooperativity limit $c \to \infty$. In our screen we use a finite cooperativity $c = 100$ to avoid the extreme case of excluding conformations with only signal bound completely; this value of c is still large enough, however, to replicate the dynamics of the neural tube network. We thus take (6.8) with $c = 100$ as the form of protein production rates in our screen; compared to the neural tube case this allows us to include both activating and repressive interactions.

Adding a protein decay term (with unit decay rate) and stochastic fluctuations, the dynamics of the three-node networks in our screen, with protein levels x_1, x_2 and x_3, is

thus described by

$$\frac{d}{dt}x_j = \frac{w_{j,\mathrm{p}} + c\, w_{j,\mathrm{p}}\,[-1 + (1 + k_{j,\mathrm{in}}x_{\mathrm{in}}/c)\phi_j]}{w_{j,\mathrm{p}} + c\, w_{j,\mathrm{p}}\,[-1 + (1 + k_{j,\mathrm{in}}x_{\mathrm{in}}/c)\phi_j] + (1 + k_{j,\mathrm{in}}x_{\mathrm{in}}/c)\phi_j + \psi_j - 1} - x_j \tag{6.11}$$

$$\phi_j = \prod_{i\neq j} (1 + [k_{ji}]_+ x_i/c)^2$$

$$\psi_j = \prod_{i\neq j} (1 + [k_{ji}]_- x_i)^2$$

for $j = 1, 2, 3$; compared to (6.8) we have dropped all tildes to unclutter the notation. We have also allowed the sets $\mathcal{P}$ and $\mathcal{N}$ of activating and repressing transcription factors to be determined implicitly by the system parameters. This is done by generalizing the affinities k_{ji} so that a positive sign indicates an activation of j by i and a negative sign a repression. The corresponding switching of species i between the products over activators and repressors is achieved mathematically by setting $[k]_+ = \max(k, 0)$ and $[k]_- = \max(-k, 0)$.

To mimic the structure of the neural tube network, we assume that only proteins 1 and 2 have direct signal inputs, while 3 does not, so that $k_{3.\mathrm{in}} = 0$. This leaves 11 network parameters: 2 for the signal (gradient) inputs from the gradient ($k_{1,\mathrm{in}}$ into node 1 and $k_{2,\mathrm{in}}$ into node 2), 6 from the interactions between TFs (k_{12}, k_{13}, k_{21}, k_{23}, k_{31} and k_{32}) and 3 for polymerase binding weights ($w_{1,\mathrm{p}}$, $w_{2,\mathrm{p}}$ and $w_{3,\mathrm{p}}$).

6.4. Parameter exploration

We explored the 11 dimensional parameter space specified above using a uniform log distribution ($\log_{10}$), where the ranges are set differently depending on the parameter. Specifically we chose the ranges as: $\mathrm{range}(k_{\mathrm{in}}) = [10 : 400]$, $\mathrm{range}(w_{\mathrm{p}}) = [0.1 : 10]$, $\mathrm{range}(k_{ji}) = [-100 : -1] \cup [1 : 100]$ with the sign of each regulation k_{ji} being chosen

randomly.

We explored parameter combinations for a three node network defined in the form (6.11). The main criterion for choosing a viable set of parameters was that they must produce a patterned steady state, i.e. a saddle-node bifurcation on the same gradient as in the neural tube: defined as $x_{\text{in}} = e^{-s/0.15}$ where s defines dorsal-ventral neural tube position and ranges from 0 to 1. To avoid trivial effects from shifts in the boundary position we set a further constraint that the bifurcation must occur at a position s in the same range as in the neural tube network, $0.165 \leq s \leq 0.17$. More specifically networks were required to be monostable below $s = 0.165$, with high levels of x_1; and bistable beyond $s = 0.17$, with one state having high x_2 and the other high x_1 (with "high" being a concentration value above 0.6). For each network meeting these criteria, we then proceeded to calculate the MAPs, as for the neural tube network (as explained in Appendix 5.2), and the jump time. We selected networks that have boundaries sharper than a certain threshold, set by requiring the boundary to be no wider than 0.2 fractional neural tube units; boundaries were calculated based on their transition time obtained from simulating the SDEs. To simulate the neural tube network from (A.1.1) in the screen we used the standard parameters from that network, reverting to the original version (Cohen et al., 2014) with maximal concentrations of unity for all TFs in order to ensure comparability with the networks produced by the screen. We removed all terms relating to Irx3, as these do not contribute substantially to the dynamics of transitioning from a pMN to a p3 steady state. We further set production and degradation rates to be equal to unity in the screen as these simply scale the jump time and do not affect the results.

In analysing the results of the network screen we quantified the curvature of the MAP as the largest perpendicular distance of any point on the MAP from the straight line between steady state and transition point, normalised by the total length of this line.

We refer to this value throughout the text by the shorthand "curvature" as it gives a quantitative indication of how much the MAP deviates from the shortest path. The curvature was measured at $s = 0.25$ and the robustness of the results with respect to this choice of neural tube position was tested by comparing with multiple other locations, with qualitatively similar results in all cases (data not shown).

In the analysis we also characterised networks by the strength of the contribution of the third node, which does not receive direct signal input. We quantified this by taking the value of x_3 at the steady state and transition point (saddle point) and multiplying each by parameters for the repression or activation of nodes 1 and 2 by node 3, taking the maximum value. The multiplication by representative concentration levels of the third node was motivated by the fact that when those concentrations are small, even large interaction parameter values have small net effects.

Networks with a low third node contribution are effectively two node networks, and turned out to have low MAP curvature. This led us to explore other mechanisms for generating sharp boundaries. Geometrically, in the space of expression levels, the speed at which the steady state and saddle point separate as a function of neural tube position s is a plausible contributor to boundary sharpness because even if the fluctuations around the initial steady state favour a jump, such a jump will be inhibited by a large separation between steady state and transition point. High separation speed should thus lead to rapidly increasing jump times and hence to sharp boundaries. To measure separation speed we focussed on a fixed position (chosen as $s = 0.25$) along the neural tube, beyond the saddle-node bifurcation, and calculate the Euclidean distance between steady state and transition point. We then used this as a simple quantitative indication of separation speed. We checked the robustness of this measure by performing the measurement for different *fixed* positions along the neural tube, and also at *variable* locations chosen as the centre of the boundary region for each network; we found qualitatively similar results

in every case (data not shown).

When a network had a high separation speed, this typically resulted in the steady state (the expression profile) of x_2 varying, i.e. changing within a domain of the steady state pattern. We quantified this heterogeneity by the standard deviation of x_2 within the region of high x_2 expression. This confirmed (see Fig. 6.1) that sharp 2D networks have a higher level of heterogeneity than 3D networks, which use the curvature of the MAP to generate sharpness.

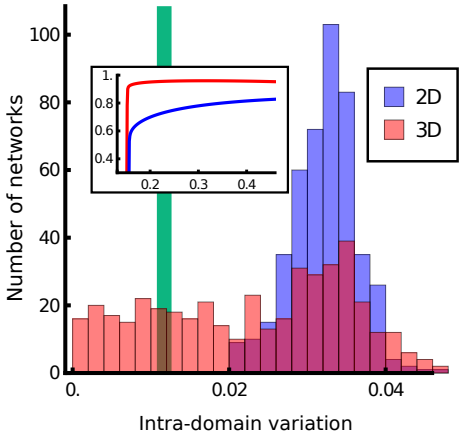


Figure 6.1: Histogram of variation of the expression level of the second node within its domain of expression for 3D (red) and 2D (blue) networks; inset shows example variation of expression levels across a domain. 3D networks can generate domains of expression with more constant levels of expression (lower domain variation) than 2D networks, which rely on separation speed to create sharp boundaries. Green line represents the WT network.

Analysing the networks

For the networks recovered from the screen, we compared the boundary precision with the extent its MAP deviates from the shortest path to the transition (we informally refer to this quantity as "curvature"). This showed a positive correlation, consistent with our observations in the WT network, of high curvature and low boundary width. This supported the idea that the shape of the transition pathway contributes to boundary precision (Fig. 6.2C). Nevertheless, within the screen, for any given level of boundary

sharpness, there were a range of MAP curvature values. We therefore investigated additional features that might affect boundary precision. We found that a subset of the networks do not rely on path curvature to achieve precision and instead functioned effectively as two node networks (Fig. 6.2D). For these networks, the major contributor to boundary precision was the rate at which the steady state and transition point separated in response to changes in level of the input signal: the higher the rate of separation, the sharper the boundary (Fig. 6.2B). We termed this "separation speed". Plotting both curvature and separation speed for the networks recovered from the screen indicated that both features contribute to precision (Fig. 6.2E). Moreover the most precise boundaries were generated by networks that exploited both separation speed and curvature, which includes the Pax6-Olig2-Nkx2.2 network (Fig. 6.2E-F).

An important corollary, for networks in which separation speed dominates, is that within the region in which gene 2 (x_2, analogous to Olig2 in WT) is expressed, the level of its expression changes markedly. This produces a graded expression domain rather than a uniform domain and precludes the generation of blocks of progenitors with constant levels of gene expression (Fig. 6.1). By contrast, networks that rely on curvature to achieve a sharp boundary allow uniform gene expression levels within the gene expression domain as the steady state does not need to change its value dramatically (Fig. 6.1). This is explained in more detail in the following section, using the example of the neural tube.

6.5. Characterisation of topologies

We then analysed the topologies of the networks resulting from the screen. To sort networks into topologies we used thresholds to identify whether nodes 1 and 2 receive significant signal input, and for each of the TF nodes whether it significantly activates

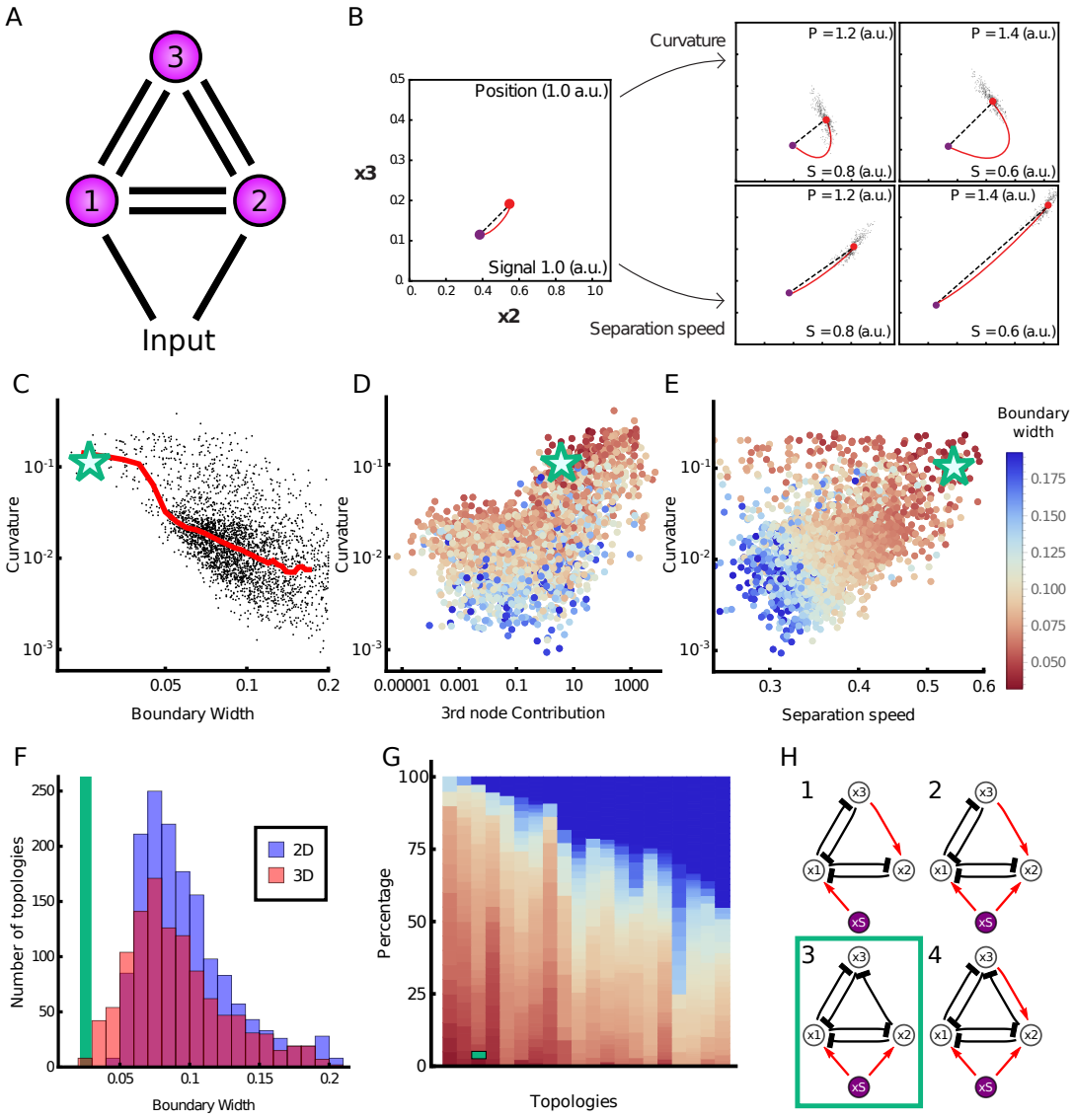

A
3
1
2
Input
B
Position (1.0 a.u.)
Signal 1.0 (a.u.)
x3
x2
Curvature
Separation speed
P =1.2 (a.u.)
P =1.4 (a.u.)
S =0.8 (a.u.)
S =0.6 (a.u.)
C
Curvature
Boundary Width
D
Curvature
3rd node Contribution
E
Curvature
Separation speed
Boundary width
F
Number of topologies
Boundary Width
2D
3D
G
Percentage
Topologies
H
1
2
3
4
x1
x2
x3
xS

Figure 6.2: A computational screen identifies three-node networks that produce sharp boundaries. (**A**) Three node networks, comprising all possible interactions and a morphogen input into two nodes, were screened to identify those producing a sharp boundary. (**B**) Two mechanisms for producing a rapid change in jump time, and consequently a precise boundary, in response to signal changes. Close to the boundary (Position 1.0 a.u.; Signal 1.0 a.u.) the steady state (red point) is near the transition point (purple point) in gene expression space. Further away (increasing Position; decreasing Signal) curvature of the MAP (red line) with respect to the shortest pathway (top row) or the rate at which the steady state separates from the transition point (bottom row) can contribute to increasing boundary precision. (**C**) For each of the networks recovered from the screen (points), the boundary width was compared to the deviation of the MAP from the shortest path to the transition (curvature). The red line indicates median value and illustrates that sharper boundaries (smaller width) tend to have higher MAP curvature. The green star represents the WT neural tube network, highlighting that this network performs as expected with respect to the screen. (**D**) A plot of curvature compared to the effective contribution of the third node in the network indicates that networks with little contribution from third node can produce sharp boundaries (boundary width indicated by colour of the point, colour is consistent throughout **D** & **G**). Green star represents the WT network. (**E**) Curvature compared to separation speed: the rate at which the steady state and transition point separate in response to decreasing signal. Colour of points by boundary width indicates both high curvature and high separation speed contribute to the sharpest boundaries. Green star represents the WT network. (**F**) Histogram of boundary width in 3D (red) and 2D (blue) networks. 3D networks have the potential to achieve greater sharpness (lower boundary width). Green line represents the WT network. (**G**) The most common topologies, arranged in order of fraction of networks with precise boundaries; each column represents an individual topology. Dark blue indicates networks with a wider boundary. Topologies are shown in Fig. 6.3. Green line represents the WT network (**H**) Four topologies that favour the sharpest boundaries, according to the ordering of **G**; they all generate networks with MAPs that have high curvature. These networks comprise inhibition from node 2 to node 3, and lack repression from node 3 to node 2. The WT neural tube network is represented in topology 3.

or represses the other TFs. Starting with the former, within the input parameter range $[10 : 400]$ for nodes 1 and 2, we took any parameter $30 < k_{\mathrm{in}}$ to be a positive input; lower values were classified as lack of input. This cutoff was chosen by testing a range of different values and imposing the constraints that neither classified the majority of networks as having two inputs (which would provide no information on the input topology, as could happen if the cutoff was too low) nor assign any network to a topology with no inputs (which would not make biological sense and would occur when the cutoff is too high). For interactions between nodes we took into account not only the parameters k_{ji} but whether each parameter in conjunction with the actual states of the system would have a noticeable effect. We evaluated interactions by considering the contribution of an interaction given the highest level that the effector node can take. Accordingly, we consider an interaction with $0.3 < |k_{ji}| \max(x_i)$ to be significant, otherwise we classify it as negligible. The maximum was taken over all steady states for all neural tube positions. The cutoff value of 0.3 was chosen by systematic inspection of a representative number of networks, for which we compared the dynamics with and without individual interactions and assessed whether these were qualitatively identical or not. To assess the robustness of the cutoff value, we varied it within a range up to an order of magnitude larger and found that the results of our characterisation of network topologies remained qualitatively the same (data not shown).

With this approach we classified all the 3D network parameter sets into topologies, determined those that occurred most often (Fig. 6.3) and plotted the boundary precisions they generate (Fig. 6.2H). The results indicated that although some topologies are more frequently represented amongst networks producing a sharp boundary, there is no single topology that ensures sharpness. Some networks (such as 1–4 in Fig. 6.3) prevented the boundary from becoming very imprecise, but even within these network topologies the range of sharpness was large (Fig. 6.2G,H & Fig. 6.3 & Fig. 6.4). This leads to

the conclusion that the dynamical properties generated by the network, rather than the structure of the network determines boundary precision. Indeed, we confirmed by analysing each topology separately that the main indicators of sharpness are the two mechanisms identified in the main text: curvature of transition path and separation speed (Fig. 6.4).

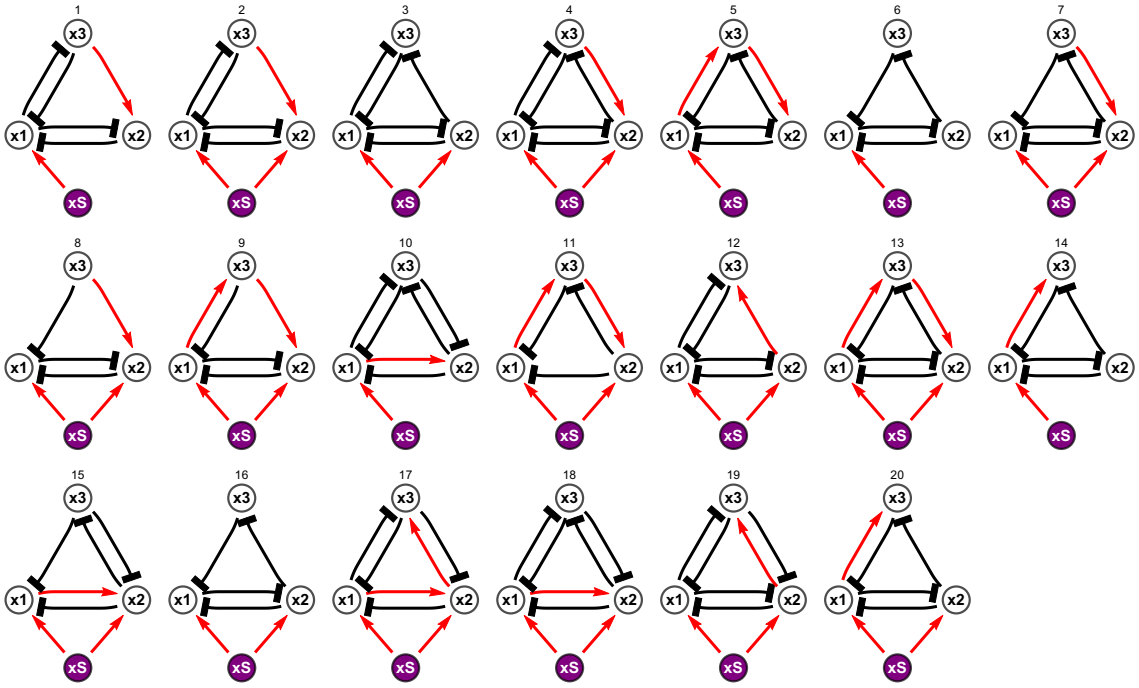


Figure 6.3: List of topologies that generate sharp boundaries, sorted in the same order as Fig. 6.2**H**. Red arrows indicate activation, black lines with blunt ends represent repression. Mutual repression between the first and second nodes (1 and 2) is a consistent feature, as well as the input from the signal to the first node. For the sharpest networks, a mutual repression between the first and third nodes is observed.

Finally, we assessed whether particular network topologies nonetheless favoured boundary sharpness. Many topologies were able to generate sharp boundaries (Fig. 6.2G,H), consistent with the expression dynamics produced by the network being key to determining behaviour. Nevertheless, four topologies appeared to be most effective at preventing very imprecise boundaries (numbered 1-4 in Fig. 6.2H). These tended to have similar separation speeds but much higher curvature than the networks with other topologies

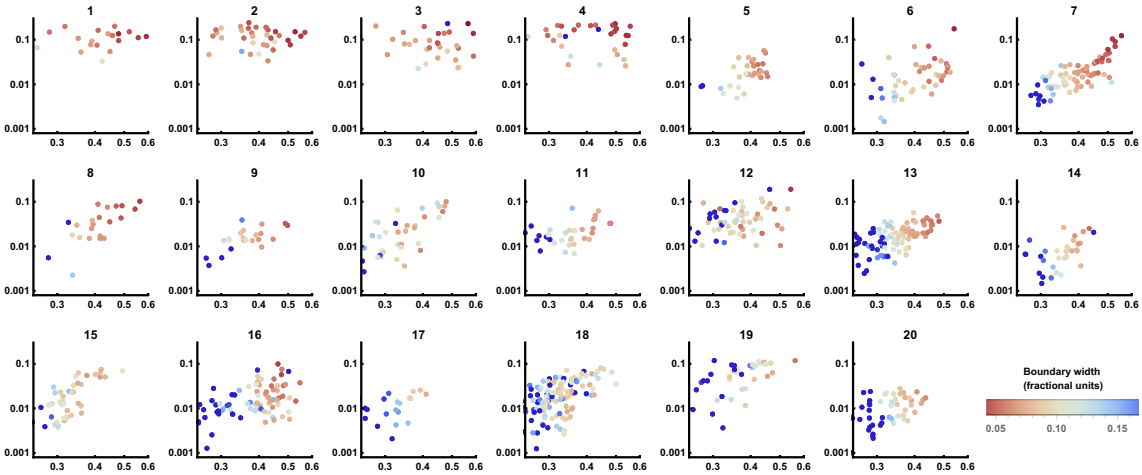


Figure 6.4: MAP curvature (x axis) plotted against separation speed (y axis) with boundary width indicated by colouring. The data are equivalent to those shown in Fig. 6.2**E**, using the same data axes and ranges, but here each plot represents an individual network topology and networks with wide boundaries have been included in the plots (deep blue). Network topologies are ordered as in Fig. 6.3. While separation speed does not exhibit obvious differences between topologies, network topologies 1–4 have consistent high curvature.

(Fig. 6.4). Crucial for this behaviour was the inhibition of gene 3 by gene 2, and the absence of repression of gene 2 by gene 3 (Fig. 6.2G & 6.3). This regulatory configuration generates curvature by allowing a steep decrease in x_3, the concentration of gene 3, while sustaining high levels of x_2 prior to the transition. Notably, the WT neural tube network conforms to this configuration, raising the possibility that it was adopted in the developing vertebrate neural tube for its capacity to generate precise patterns in the presence of intrinsic noise. Moreover, the graded expression within the domain associated with the use of separation speed (Fig. 6.4) is represented by Pax6 (x_3) and this allows constant levels of Olig2 (x_2; the gene necessary for defining neuronal subtype identity) within the progenitor domain. Hence, an understanding of the dynamical properties of the GRN offers an explanation for its structure and the resulting gene expression behaviour.

Overall this screen uncovered fundamental principles that led to a precise boundary in response to externally imposed positional information: a rapid change in transition time between steady states in response to a change in signal. We have further identified the

two main mechanisms for this: a large curvature in the MAP that results from having an additional node, or the steady state and saddle point pulling apart at a rapid rate. The curvature is particularly interesting, as it can only be obtained by networks with three or more nodes because this additional dimension provides a space into which fluctuate away from the saddle point. Moreover, the overrepresentation of certain network structures suggest that specific network topologies are more suited to generating sharp boundaries.

7

Concluding statements and future perspectives

In this chapter we summarise the contributions of this thesis, their importance to science as well as future avenues of research that these contributions open up.

7.1. Projection methods and memory functions

In this thesis we have derived two methods to reduce the dimensionality of a system of ODEs. They are both based on the Zwanzig-Mori formalism (Mori, 1965, Nakajima, 1958, Zwanzig, 1961) and thus allow an arbitrary choice of subnetwork and bulk. The first method (Chapter 2 & 3) has been derived specifically for Gene Regulatory Network (GRN) systems and requires the system to be expanded around a steady state. The memory functions correct the dynamics of the system to take into account the effect of the bulk species. This method is exact when close to steady state and the bulk species are initialised at steady state. The second method (Chapter 4) has been derived generally for any set of ODEs and uses the baseline approximation that the bulk species are in Quasi-Steady State (QSS). The memory functions then act to correct for this assumption. As a consequence this method is exact in the limit where the dynamics in the bulk are infinitely fast. Together these two methods provide a comprehensive toolkit to approach GRN and other systems by reducing their dimensionality in a very flexible way. Furthermore, the second method has the advantage of being extremely accurate in its timecourse predictions even for strongly nonlinear systems.

A further contribution is the interpretation of a system that the Zwanzig-Mori methods derived in this thesis can provide (Chapter 2-4). As shown in Chapter 3, the steady state expansion method provides relatively simple memory functions that can be easily interpreted to understand what elements of a system contribute towards generating a steady state. This analysis led to understanding what elements of a network may provide robustness to a pattern. We then extended this analysis using nonlinear memory functions in Chapter 4, and confirmed the findings of Chapter 3. The advantage of analysing these memory functions is that we can observe the decomposition of the memory functions far from steady state. We show that when combining both approaches it is possible

to determine whether interactions are generally weak or if they are at QSS. With these results we can understand in detail the effect of individual parts of a network in an analytical way, providing insights that are not easily obtainable otherwise, e.g. by direct numerical solution of the full dynamics.

7.2. Precision of gene expression boundaries

To understand the accuracy and precision of cell fates decision, a SDE model was generated based on previous work (Chapter 5) & (Cohen et al., 2014). This model was used to investigate boundary precision between gene expression domains in response to a gradient of morphogen signal in the vertebrate ventral neural tube. The model was able to accurately reproduce experimental data on the p3 and pMN domains of the WT neural tube as well as previously reported phenotypes. The model recapitulated the effect of loss of sharpness in the Pax6$^{-/-}$ and provides an explanation to this: the transition time from one steady state to another is different depending on neural tube position; this time changes from very high to very low very quickly in the WT but not so in the Pax6$^{-/-}$. Motivated by the model, an additional mutant was analysed, namely a deletion of an Olig2 enhancer that contains multiple binding sites for several of the transcription factors in the model. We modelled this by modifying the weight of interactions within the system. This mutation also produced a reduced precision in boundary, though less dramatic than the Pax6$^{-/\cdot}$, consistent with the experimental observations. The model again provides an explanation for this mutant's loss of boundary precision, following the same reasoning as for the Pax6$^{-/-}$.

To obtain an understanding of why the jump time changes at different speeds across the neural tube we calculated and analysed the Minimum Action Path (MAP). This suggested that the key property leading to rapid changes in transition time is the direction of fluctuations near the initial steady state (curvature). When these fluctuations are *not*

pointing towards the transition point in phase space, as the steady state and transition point separate, the probability of a jump taking place is reduced dramatically. However, when the fluctuations *are* pointing towards the transition point, this probability of a jump is higher; the jump time is then lower and consequently also varies at a slower speed across the neural tube. This provides a unifying explanation for both mutant phenotypes: the Pax6$^{-/-}$ loses the dimension in which the fluctuations point away from the transition point and thus has the most dramatic effect; the Olig2 enhancer mutant simply reorients the direction of the fluctuations to be slightly more aligned towards the saddle node. To allow us to quantify the directionality of the fluctuations we design a measurable quantity that we call "curvature", as fluctuations pointing away from a transition point produce a curve in the corresponding MAP. To our knowledge this is the first time such a mechanism has been proposed, and through the experiments of Katherine Exelby we have been able to provide solid evidence that it is present in a well-understood biological system.

Motivated by these findings, we went on to investigate the generality of the mechanism for boundary precision involving fluctuations around a steady state (Chapter 6). We performed a computational screen of networks involving three nodes by randomly exploring the parameter space of interactions, input signals and basal production for each gene. We find that in addition to the curvature, the speed at which the initial steady state and transition point separate in response to signal is of fundamental importance, we term this property "separation speed". Both the curvature and separation speed may affect substantially the change in transition time, which in turn determines how precise a boundary between gene expression domains will be. We then classify the resulting networks into topologies based on the weight of the regulations within a network as well as input strength. An initial finding from this is that, regardless of the topology of a system, the main indicators for whether a network will produce a sharp boundary are its

curvature and separation speed. We highlight a particular set of topologies that always present a high curvature; this property then ensures that these topologies always present a minimum amount of precision. An interesting finding is that the neural tube network from Chapter 5 belongs to one of these topologies and also presents a high amount of curvature and separation speed. The results from this chapter highlight the importance of understanding the dynamical properties of a system that can be more informative than the network topology of such systems, as has been suggested in other studies (Ingram et al., 2006, Jiménez et al., 2017).

7.3. Future perspectives

7.3.1. Memory functions

In regards to the dimensionality reduction methods derived in this thesis, there are multiple avenues that could be pursued. One is to investigate how to interpret the memory functions generated by channels that are shown to contribute substantially to the dynamics of a system, and how each channel of memory contributes to the dynamics of the system. Large contributions imply that the interactions that encode such channels are not close to the limit of behaving infinitely fast, and it is thus important for the original dynamics that these links represent a slow response to concentrations of other elements in the system. From a biological perspective it would be interesting to understand if this provides an understanding of how early a cell fate decision is made, and if this is encoded in the memory through specific interactions.

As we have shown that our two projection approaches can work in a complimentary way to each other it would likely be beneficial to develop further projection approaches with different underlying assumptions. This could lead to memory functions that capture some of the same properties as those derived in this thesis but also further features

dependent on the assumptions made. One could also obtain memory functions resulting from equations similar to those derived in (Gouasmi et al., 2017), where a more drastic approximation of signal propagated from the initial conditions is performed. This kind of work can then be compared and contrasted to the approaches presented in this thesis to understand its advantages and limitations.

A natural connection between the two main themes of this thesis would be to derive projection approaches that account for stochasticity in the form of intrinsic noise. This would require one to derive the explicit form of the random force terms from (2.9) or (4.7) in a similar way to (Bravi and Sollich, 2017a, Bravi et al., 2016). Deriving such terms could allow for a rigorous investigation of the influence of intrinsic noise in establishing the timescale of dynamical systems. The memory functions could be compared across systems with different levels of noise to understand the influence of the noise on transients, oscillations or choice of steady states. A result of this type, similar to that of Chapter 4, could apply to rather general dynamical systems and may not be limited to a specific biological context.

7.3.2. Boundary precision

To improve the parametrisation of the model it would be of great use to better quantify the noise of gene expression experimentally. This would lead to a tightening of the range of our noise parameter and increase the predictive ability of the model. Following such measurements, a natural extension of the neural tube model would be to observe the effect of fluctuations in signalling from the morphogen gradient over time. This would introduce an additional source of noise that would be expected to affect the behaviour of the system. It has previously been observed that intrinsic noise has the capacity to correct for noise in the signal (Zhang et al., 2012), but it is not obvious if this would be the case in our system. In pursuing such an investigation it would be

important to properly characterise the noise distribution of the signal. This could be done either by modelling the signalling components of (Cohen et al., 2015) stochastically, or modelling the signal with different types of stochastic processes such as Ornstein-Uhlenbeck, Poisson or Brownian motion. Depending on the chosen noise form the behaviour of the resulting system could vary substantially, therefore this choice has to be correctly motivated biologically. It would then be of great interest to perform a similar analysis in a generalised computational screen; one could use the same networks we have found in this thesis and understand if curvature and separation speed are still key factors for boundary precision in this new context. It would also be interesting to understand how these two properties behave in the context of a noisy signal and if there are additional factors that are relevant for boundary sharpness.

Regarding modifications to the signal, one specific form that is motivated by biological data is a change in signalling levels over time. In the neural tube, the levels of Gli activity (the downstream transcriptional effector for Sonic HedgeHog) produce a transient behaviour *i.e.* levels of Gli rise and then fall (Balaskas et al., 2012, Cohen et al., 2015, Junker et al., 2014). While understanding the causes of this behaviour, for example on the basis of adaptation, is in itself an interesting problem (Cohen et al., 2015), it would also be fascinating to understand what effect this transient in signalling levels has on the patterning of the neural tube. It is likely that this process allows the system to "move" through the phase space quickly and avoid regions with low drift that could otherwise slow down the patterning process. It is unclear what effect the signalling dynamics in the signal will have in the presence of intrinsic noise in the network, and this would be important to understand as several other developmental signalling pathways present this type of behaviour.

From a theoretical aspect there are still further considerations that would be worthwhile regarding the transition between steady states. In this work we have specifically

looked for saddle-node bifurcations, which lead from monostability to bistability. While these tend to be the most common for our kind of system it is possible to have further complexity with similar networks, such as systems with tristability or damped oscillatory dynamics (De Mot et al., 2016, Perez-Carrasco et al., 2018). It would be important to find out how likely it is for such a system to occur in parameter space and then to understand what effect this may have on transitions between steady states. In addition to this, another mechanism through which a sharp boundary could be achieved is if a system drastically minimises the size of the bistable region. Our preliminary observations (data not shown) indicated that this is a rare occurrence but it could provide an interesting case study, especially with regards to the question of whether the properties of separation speed and curvature are also optimised in such a system.

Finally, it would be interesting to analyse whether the mechanism for generating precision in a boundary that we have described for the neural tube in Chapter 5 is found in other systems. The results from Chapter 6 emphasise that these behaviours are not unique to the neural tube network and indeed are likely to be found elsewhere. An obvious system for such an analysis is given by the *Drosophila melanogaster* GAP genes (Jaeger, 2011). The similarities between that network and the neural tube network have already been highlighted by other groups (as discussed in Section 1.5.4 of Chapter 1, (Briscoe and Small, 2015, Verd et al., 2019)). It would be important to understand through both theoretical analysis and experimental perturbations to the corresponding system if curvature or separation speed are present. If so, this would strongly imply that such properties may be selected through evolution in a convergent manner. It would also suggest that the mechanisms first presented in this thesis and corresponding manuscript are general properties found throughout developmental systems.

References

Alaynick, W. A., Jessell, T. M., and Pfaff, S. L. (2011). SnapShot: Spinal cord development. *Cell*, 146(1):178.e1.

Almuedo-Castillo, M., Bläßle, A., Mörsdorf, D., Marcon, L., Soh, G. H., Rogers, K. W., Schier, A. F., and Müller, P. (2018). Scale-invariant patterning by size-dependent inhibition of Nodal signalling. *Nature cell biology*, 20(9):1032–1042.

Alon, U. (2007). Network motifs: Theory and experimental approaches. *Nature Reviews Genetics*, 8(6):450–461.

Anderson, D. F. and Kurtz, T. G. (2011). Continuous Time Markov Chain Models for Chemical Reaction Networks. In *Design and Analysis of Biomolecular Circuits*, pages 3–42. Springer New York, New York, NY.

Angeli, D., Ferrell, J. E., and Sontag, E. D. (2004). Detection of multistability, bifurcations, and hysteresis in a large class of biological positive-feedback systems. *Proceedings of the National Academy of Sciences*, 101(7):1822–1827.

Apri, M., de Gee, M., and Molenaar, J. (2012). Complexity reduction preserving dynamical behavior of biochemical networks. *Journal of Theoretical Biology*, 304:16–26.

Balaskas, N., Ribeiro, A., Panovska, J., Dessaud, E., Sasai, N., Page, K. M., Briscoe, J., and Ribes, V. (2012). Gene regulatory logic for reading the sonic hedgehog signaling gradient in the vertebrate neural tube. *Cell*, 148(1-2):273–284.

Beck, C. L., Lall, S., Liang, T., and West, M. (2009). Model reduction, optimal prediction, and the Mori-Zwanzig representation of Markov chains. *Proceedings of the IEEE Conference on Decision and Control*, pages 3282–3287.

Ben-Zvi, D., Shilo, B. Z., Fainsod, A., and Barkai, N. (2008). Scaling of the BMP activation gradient in Xenopus embryos. *Nature*, 453(7199):1205–1211.

Berkooz, G., Holmes, P., and Lumley, J. L. (1993). The Proper Orthogonal Decomposition in the Analysis of Turbulent Flows. *Annual Review of Fluid Mechanics*, 25(1):539–575.

Bintu, L., Buchler, N. E., Garcia, H. G., Gerland, U., Hwa, T., Kondev, J., and Phillips, R. (2005). Transcriptional regulation by the numbers: Models. *Current Opinion in Genetics and Development*, 15(2):116–124.

Bravi, B. and Sollich, P. (2017a). Critical scaling in hidden state inference for linear Langevin dynamics. *Journal of Statistical Mechanics: Theory and Experiment*, 2017(6):063504.

Bravi, B. and Sollich, P. (2017b). Inference for dynamics of continuous variables: The extended Plefka expansion with hidden nodes. *Journal of Statistical Mechanics: Theory and Experiment*, 2017(6):063404.

Bravi, B., Sollich, P., and Opper, M. (2016). Extended Plefka expansion for stochastic dynamics. *Journal of Physics A: Mathematical and Theoretical*, 49(19):194003.

Briscoe, J. and Kicheva, A. (2017). The physics of development 100 years after D'Arcy Thompson's "On Growth and Form". *Mechanisms of Development*, 145:26–31.

Briscoe, J., Pierani, A., Jessell, T. M., and Ericson, J. (2000). A homeodomain protein code specifies progenitor cell identity and neuronal fate in the ventral neural tube. *Cell*, 101(4):435–445.

Briscoe, J. and Small, S. (2015). Morphogen rules: design principles of gradient-mediated embryo patterning. *Development*, 142(23):3996–4009.

Briscoe, J., Sussel, L., Serup, P., Hartigan-O'Connor, D., Jessell, T. M., Rubenstein, J. L. R., and Ericson, J. (1999). Homeobox gene Nkx2.2 and specification of neuronal identity by graded Sonic hedgehog signalling. *Nature*, 398(6728):622–627.

Bunin, G., Kafri, Y., and Podolsky, D. (2012). Large deviations in boundary-driven systems: Numerical evaluation and effective large-scale behavior. *EPL*, 99(2).

Cardelli, L. (2014). Morphisms of reaction networks that couple structure to function. *BMC Systems Biology*, 8(1):84.

Chizhikov, V. V. and Millen, K. J. (2004). Mechanisms of roof plate formation in the vertebrate CNS. *Nature Reviews Neuroscience*, 5(10):808–812.

Chorin, A. and Stinis, P. (2006). Problem reduction, renormalization, and memory. *Communications in Applied Mathematics and Computational Science*, 1(1):1–27.

Chorin, A. J., Hald, O. H., and Kupferman, R. (2000). Optimal prediction and the Mori-Zwanzig representation of irreversible processes. *Proceedings of the National Academy of Sciences*, 97(7):2968–2973.

Chorin, A. J., Hald, O. H., and Kupferman, R. (2002). Optimal prediction with memory. *Physica D: Nonlinear Phenomena*, 166(3-4):239–257.

Chorin, A. J., Hald, O. H., and Kupferman, R. (2006). Prediction from partial data, renormalization, and averaging. *Journal of Scientific Computing*, 28(2-3):245–261.

Chubb, J. R., Trcek, T., Shenoy, S. M., and Singer, R. H. (2006). Transcriptional Pulsing of a Developmental Gene. *Current Biology*, 16(10):1018–1025.

Clark, E. (2017). Dynamic patterning by the Drosophila pair-rule network reconciles long-germ and short-germ segmentation. *PLOS Biology*, 15(9):e2002439.

Cohen, M., Briscoe, J., and Blassberg, R. (2013). Morphogen Interpretation: The Transcriptional Logic Of Neural Tube Patterning. *Current Opinion in Genetics and Development*, 23(4):423–428.

Cohen, M., Kicheva, A., Ribeiro, A., Blassberg, R., Page, K. M., Barnes, C. P., and Briscoe, J. (2015). Ptch1 and Gli regulate Shh signalling dynamics via multiple mechanisms. *Nat Commun*, 6:6709.

Cohen, M., Page, K. M., Perez-Carrasco, R., Barnes, C. P., and Briscoe, J. (2014). A theoretical framework for the regulation of Shh morphogen-controlled gene expression. *Development*, 141(20):3868–3878.

Cotterell, J. and Sharpe, J. (2010). An atlas of gene regulatory networks reveals multiple three-gene mechanisms for interpreting morphogen gradients. *Molecular Systems Biology*, 6(425):425.

Davidson, E. H. (2001). *Genomic regulatory systems : development and evolution*. Academic Press.

de la Cruz, R., Perez-Carrasco, R., Guerrero, P., Alarcon, T., and Page, K. M. (2018). Minimum Action Path Theory Reveals the Details of Stochastic Transitions Out of Oscillatory States. *Physical Review Letters*, 120(12):128102.

De Mot, L., Gonze, D., Bessonnard, S., Chazaud, C., Goldbeter, A., and Dupont, G. (2016). Cell Fate Specification Based on Tristability in the Inner Cell Mass of Mouse Blastocysts. *Biophysical Journal*, 110(3):710–722.

Dessaud, E., McMahon, A. P., and Briscoe, J. (2008). Pattern formation in the vertebrate neural tube: a sonic hedgehog morphogen-regulated transcriptional network. *Development*, 135(15):2489–2503.

Dessaud, E., Ribes, V., Balaskas, N., Yang, L. L., Pierani, A., Kicheva, A., Novitch, B. G., Briscoe, J., and Sasai, N. (2010). Dynamic assignment and maintenance of positional identity in the ventral neural tube by the morphogen sonic hedgehog. *PLoS Biology*, 8(6):e1000382.

Diego, X., Marcon, L., Müller, P., and Sharpe, J. (2018). Key Features of Turing Systems are Determined Purely by Network Topology. *Physical Review X*, 8(2):21071.

Doetschman, T., Gregg, R. G., Maeda, N., Hooper, M. L., Melton, D. W., Thompson, S., and Smithies, O. (1987). Targetted correction of a mutant HPRT gene in mouse embryonic stem cells. *Nature*, 330(6148):576–578.

El-Sherif, E., Averof, M., and Brown, S. J. (2012). A segmentation clock operating in blastoderm and germband stages of Tribolium development. *Development (Cambridge)*, 139(23):4341–4346.

Eldar, A. and Elowitz, M. B. (2010). Functional roles for noise in genetic circuits. *Nature*, 467(7312):167–173.

Elowitz, M. B. and Leibier, S. (2000). A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767):335–338.

Ericson, J., Rashbass, P., Schedl, A., Brenner-Morton, S., Kawakami, A., Van Heyningen, V., Jessell, T. M., and Briscoe, J. (1997). Pax6 controls progenitor cell identity and neuronal fate in response to graded Shh signaling. *Cell*, 90(1):169–180.

Exelby, K., Herrera-Delgado, E., Perez, L. G., Perez-Carrasco, R., Sagner, A., Metzis,

V., Sollich, P., and Briscoe, J. (2019). Precision of Tissue Patterning is Controlled by Dynamical Properties of Gene Regulatory Networks. *bioRxiv*.

Feliu, E. and Wiuf, C. (2013). Simplifying biochemical models with intermediate species. *Journal of the Royal Society Interface*, 10(87):20130484–20130484.

Fick, A. (1855). Ueber Diffusion. *Annalen der Physik und Chemie*, 170(1):59–86.

Froyland, G., Gottwald, G. A., and Hammerlindl, A. (2014). A Computational Method to Extract Macroscopic Variables and Their Dynamics in Multiscale Systems. *SIAM Journal on Applied Dynamical Systems*, 13(4):1816–1846.

Garcia, H. G., Sanchez, A., Kuhlman, T., Kondev, J., and Phillips, R. (2010). Transcription by the numbers redux: Experiments and calculations that surprise. *Trends in Cell Biology*, 20(12):723–733.

Gay, S., Soliman, S., and Fages, F. (2010). A graphical method for reducing and relating models in systems biology. *Bioinformatics*, 26(18).

Gierer, A. and Meinhardt, H. (1972). A theory of biological pattern formation. *Kybernetik*, 12(1):30–39.

Gilbert, S. F. (2016). *Developmental Biology*. Sinauer, 11th edition.

Gillespie, D. T. (2000). Chemical Langevin equation. *Journal of Chemical Physics*, 113(1):297–306.

Gillespie, D. T. (2002). The Chemical Langevin and Fokker—Planck Equations for the Reversible Isomerization Reaction †. *The Journal of Physical Chemistry A*, 106(20):5063–5071.

Gonze, D., Bernard, S., Waltermann, C., Kramer, A., and Herzel, H. (2005). Spontaneous synchronization of coupled circadian oscillators. *Biophysical Journal*, 89(1):120–129.

Gotze, W. and Sjogren, L. (1992). Relaxation processes in supercooled liquids. *Reports on Progress in Physics*, 55(3):241–376.

Gouasmi, A., Parish, E. J., and Duraisamy, K. (2017). A priori estimation of memory effects in reduced-order models of nonlinear systems using the Mori-Zwanzig formalism. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 473(2205):20170385.

Gouti, M., Tsakiridis, A., Wymeersch, F. J., Huang, Y., Kleinjung, J., Wilson, V., and Briscoe, J. (2014). In vitro generation of neuromesodermal progenitors reveals distinct roles for wnt signalling in the specification of spinal cord and paraxial mesoderm identity. *PLoS Biology*, 12(8):e1001937.

Green, J. B. A. and Sharpe, J. (2015). Positional information and reaction-diffusion: two big ideas in developmental biology combine. *Development*, 142(7):1203–1211.

Grima, R., Thomas, P., and Straube, A. V. (2011). How accurate are the nonlinear chemical Fokker-Planck and chemical Langevin equations? *Journal of Chemical Physics*, 135(8):084103–084103–16.

Gugercin, S. and Antoulas, A. C. (2004). A survey of model reduction by balanced truncation and some new results. *International Journal of Control*, 77(8):748–766.

Gunawardena, J. (2010). Models in Systems Biology: The Parameter Problem and the Meanings of Robustness. *Elements of Computational Systems Biology*, pages 21–47.

Hammar, P., Walldén, M., Fange, D., Persson, F., Baltekin, Ö., Ullman, G., Leroy, P., and Elf, J. (2014). Direct measurement of transcription factor dissociation excludes

a simple operator occupancy model for gene regulation. *Nature Genetics*, 46(4):405–408.

Hijón, C., Español, P., Vanden-Eijnden, E., and Delgado-Buscalioni, R. (2010). Mori-Zwanzig formalism as a practical computational tool. *Faraday Discuss.*, 144:301–322.

Hinman, V. F., Nguyen, A. T., Cameron, R. A., and Davidson, E. H. (2003). Developmental gene regulatory network architecture across 500 million years of echinoderm evolution. *Proceedings of the National Academy of Sciences of the United States of America*, 100(23):13356–61.

Ingram, P. J., Stumpf, M. P., and Stark, J. (2006). Network motifs: Structure does not determine function. *BMC Genomics*, 7:1–12.

Inomata, H., Shibata, T., Haraguchi, T., and Sasai, Y. (2013). Scaling of dorsal-ventral patterning by embryo size-dependent degradation of spemann's organizer signals. *Cell*, 153(6):1296–311.

Jaeger, J. (2011). The gap gene network. *Cellular and Molecular Life Sciences*, 68(2):243–274.

Jaeger, J. and Monk, N. (2014). Bioattractors: Dynamical systems theory and the evolution of regulatory processes. *Journal of Physiology*, 592(11):2267–2281.

Jaeger, J. and Monk, N. (2015). Everything flows: A process perspective on life. *EMBO reports*, 16(9):1064–7.

Jaeger, J., Surkova, S., Blagov, M., Janssens, H., Kosman, D., Kozlov, K. N., Manu, Myasnikova, E., Vanario-Alonso, C. E., Samsonova, M., Sharp, D. H., and Reinitz, J. (2004). Dynamic control of positional information in the early Drosophila embryo. *Nature*, 430(6997):368–371.

Jeong, J. and McMahon, A. P. (2005). Growth and pattern of the mammalian neural tube are governed by partially overlapping feedback activities of the hedgehog antagonists patched 1 and Hhip1. *Development (Cambridge, England)*, 132(1):143–54.

Jiménez, A., Cotterell, J., Munteanu, A., and Sharpe, J. (2017). A spectrum of modularity in multi-functional gene circuits. *Molecular Systems Biology*, 13(4):925.

Junker, J. P., Peterson, K. A., Nishi, Y., Mao, J., McMahon, A. P., and van Oudenaarden, A. (2014). A predictive model of bifunctional transcription factor signaling during embryonic tissue patterning. *Developmental Cell*, 31(4):448–460.

Kærn, M., Elston, T. C., Blake, W. J., and Collins, J. J. (2005). Stochasticity in gene expression: From theories to phenotypes. *Nature Reviews Genetics*, 6(6):451–464.

Kalmar, T., Lim, C., Hayward, P., Muñoz-Descalzo, S., Nichols, J., Garcia-Ojalvo, J., and Arias, A. M. (2009). Regulated fluctuations in Nanog expression mediate cell fate decisions in embryonic stem cells. *PLoS Biology*, 7(7):e1000149.

Kicheva, A., Bollenbach, T., Ribeiro, A., Pérez Valle, H., Lovell-Badge, R., Episkopou, V., and Briscoe, J. (2014). Coordination of progenitor specification and growth in mouse and chick spinal cord. *Science*, 345(6204):1254927–1254927.

Kleinert, H. (2009). *Path Integrals In Quantum Mechanics, Statistics, Polymer Physics, And Financial Markets (5th Edition)*. World Scientific Publishing Company, 5th edition.

Kolmogorov, A. N. (1937). Étude de l'équation de la diffusion avec croissance de la quantité de matière et son application à un problème biologique. *Bull. Univ. Moskow, Ser. Internat., Sec. A*, 1:1–25.

Kondo, S. and Miura, T. (2010). Reaction-diffusion model as a framework for understanding biological pattern formation. *Science*, 329(5999):1616–1620.

Koseska, A., Zaikin, A., García-Ojalvo, J., and Kurths, J. (2007). Stochastic suppression of gene expression oscillators under intercell coupling. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 75(3):1–9.

Kutejova, E., Sasai, N., Shah, A., Gouti, M., and Briscoe, J. (2016). Neural Progenitors Adopt Specific Identities by Directly Repressing All Alternative Progenitor Transcriptional Programs. *Developmental Cell*, 36(6):639–653.

Lemons, D. S. and Gythiel, A. (1997). Paul Langevin's 1908 paper "On the Theory of Brownian Motion" ["Sur la théorie du mouvement brownien," C. R. Acad. Sci. (Paris) 146 , 530-533 (1908)]. *American Journal of Physics*, 65(11):1079–1081.

Leon, M., Woods, M. L., Fedorec, A. J., and Barnes, C. P. (2016). A computational method for the investigation of multistable systems and its application to genetic switches. *BMC Systems Biology*, 10(1):1–12.

Levine, M. and Davidson, E. H. (2005). Gene regulatory networks for development. *Proceedings of the National Academy of Sciences of the United States of America*, 102(14):4936–42.

Lewis, J. (2003). Autoinhibition with transcriptional delay: A simple mechanism for the zebrafish somitogenesis oscillator. *Current Biology*, 13(16):1398–1408.

Li, C., Cesbron, F., Oehler, M., Brunner, M., and Höfer, T. (2018). Frequency Modulation of Transcriptional Bursting Enables Sensitive and Rapid Gene Regulation. *Cell Systems*, 6(4):409–423.e11.

Li, Z., Bian, X., Caswell, B., and Karniadakis, G. E. (2014). Construction of dissipative particle dynamics models for complex fluids via the Mori-Zwanzig formulation. *Soft Matter*, 10(43):8659–8672.

Lucas, T., Tran, H., Perez Romero, C. A., Guillou, A., Fradin, C., Coppey, M., Walczak, A. M., and Dostatni, N. (2018). 3 Minutes To Precisely Measure Morphogen Concentration. *PLoS Genetics*, 14(10):e1007676.

Ma, W., Trusina, A., El-Samad, H., Lim, W. A., and Tang, C. (2009). Defining Network Topologies that Can Achieve Biochemical Adaptation. *Cell*, 138(4):760–773.

Maamar, H. and Dubnau, D. (2005). Bistability in the Bacillus subtilis K-state (competence) system requires a positive feedback loop. *Molecular Microbiology*, 56(3):615–624.

Maamar, H., Raj, A., and Dubnau, D. (2007). Noise in gene expression determines cell fate in Bacillus subtilis. *Science (New York, N.Y.)*, 317(5837):526–9.

Marcon, L., Diego, X., Sharpe, J., and Müller, P. (2016). High-throughput mathematical analysis identifies turing networks for patterning with equally diffusing signals. *eLife*, 5(APRIL2016).

McAdams, H. H. and Arkin, A. (1997). Stochastic mechanisms in gene expression. *Proceedings of the National Academy of Sciences*, 94(3):814–819.

Metzis, V., Steinhauser, S., Pakanavicius, E., Gouti, M., Stamataki, D., Ivanovitch, K., Watson, T., Rayon, T., Mousavy Gharavy, S. N., Lovell-Badge, R., Luscombe, N. M., and Briscoe, J. (2018). Nervous System Regionalization Entails Axial Allocation before Neural Differentiation. *Cell*, 175(4):1105—-1118.e17.

Mogilner, A., Wollman, R., and Marshall, W. F. (2006). Quantitative Modeling in Cell Biology: What Is It Good for? *Developmental Cell*, 11(3):279–287.

Mori, H. (1965). Transport, Collective Motion, and Brownian Motion. *Progress of Theoretical Physics*, 33(3):423–455.

Morohashi, M., Ohashi, Y., Tani, S., Ishii, K., Itaya, M., Nanamiya, H., Kawamura, F., Tomita, M., and Soga, T. (2007). Model-based Definition of Population Heterogeneity and Its Effects on Metabolism in Sporulating Bacillus subtilis. *The Journal of Biochemistry*, 142(2):183–191.

Murray, J. D. (2007). *Mathematical Biology 1: An Introduction*. Springer, 3rd edition.

Nakajima, S. (1958). On Quantum Theory of Transport Phenomena. *Progress of Theoretical Physics*, 20(6):948–959.

Naoki, H., Akiyama, R., Sari, D. W. K., Ishii, S., Bessho, Y., and Matsui, T. (2019). Noise-resistant developmental reproducibility in vertebrate somite formation. *PLoS Computational Biology*, 15(2):1–19.

Nicolis, G. (1980). *Systems Far from Equilibrium*, volume 132 of *Lecture Notes in Physics*. Springer, Berlin, Heidelberg.

Nishi, Y., Zhang, X., Jeong, J., Peterson, K. A., Vedenko, A., Bulyk, M. L., Hide, W. A., and McMahon, A. P. (2015). A direct fate exclusion mechanism by Sonic hedgehog-regulated transcriptional repressors. *Development*, 142(19):3286–3293.

Novitch, B. G., Chen, A. I., and Jessell, T. M. (2001). Coordinate regulation of motor neuron subtype identity and pan-neuronal properties by the bHLH repressor Olig2. *Neuron*, 31(5):773–789.

Oates, A. C., Morelli, L. G., and Ares, S. (2012). Patterning embryos with oscillations: structure, function and dynamics of the vertebrate segmentation clock. *Development*, 139(4):625–639.

Okino, M. S. and Mavrovouniotis, M. L. (1998). Simplification of mathematical models of chemical reaction systems. *Chemical Reviews*, 98(2):391–408.

Oosterveen, T., Kurdija, S., Alekseenko, Z., Uhde, C. W., Bergsland, M., Sandberg, M., Andersson, E., Dias, J. M., Muhr, J., and Ericson, J. (2012). Mechanistic Differences in the Transcriptional Interpretation of Local and Long-Range Shh Morphogen Signaling. *Developmental Cell*, 23(5):1006–1019.

Panovska-Griffiths, J., Page, K. M., and Briscoe, J. (2013). A gene regulatory motif that generates oscillatory or multiway switch outputs. *Journal of the Royal Society Interface*, 10(79):20120826.

Papalopulu, N. and Kintner, C. (1996). A posteriorising factor, retinoic acid, reveals that anteroposterior patterning controls the timing of neuronal differentiation in Xenopus neuroectoderm. *Development*, 122(11):3409–3418.

Perez-Carrasco, R., Barnes, C. P., Schaerli, Y., Isalan, M., Briscoe, J., and Page, K. M. (2018). Combining a Toggle Switch and a Repressilator within the AC-DC Circuit Generates Distinct Dynamical Behaviors. *Cell Systems*, 6(4):521–530.e3.

Perez-Carrasco, R., Guerrero, P., Briscoe, J., and Page, K. M. (2016). Intrinsic Noise Profoundly Alters the Dynamics and Steady State of Morphogen-Controlled Bistable Genetic Switches. *PLoS Computational Biology*, 12(10):e1005154.

Peterson, K. A., Nishi, Y., Ma, W., Vedenko, A., Shokri, L., Zhang, X., McFarlane, M., Baizabal, J. M., Junker, J. P., van Oudenaarden, A., Mikkelsen, T., Bernstein, B. E., Bailey, T. L., Bulyk, M. L., Wong, W. H., and McMahon, A. P. (2012). Neural-specific Sox2 input and differential Gli-binding affinity provide context and positional information in Shh-directed neural patterning. *Genes and Development*, 26(24):2802–2816.

Ran, F. A., Hsu, P. D., Wright, J., Agarwala, V., Scott, D. A., and Zhang, F. (2013).

Genome engineering using the CRISPR-Cas9 system. *Nature protocols*, 8(11):2281–2308.

Raser, J. M. and O'Shea, E. K. (2005). Noise in gene expression: origins, consequences, and control. *Science (New York, N.Y.)*, 309(5743):2010–3.

Rasmussen, C. E. and Williams, C. K. I. (2004). Gaussian processes for machine learning. *International journal of neural systems*, 14(2):69–106.

Rega, G. and Troger, H. (2005). Dimension Reduction of Dynamical Systems: Methods, Models, Applications. *Nonlinear Dynamics*, 41(1-3):1–15.

Reinitz, J. and Sharp, D. H. (1995). Mechanism of eve stripe formation. *Mechanisms of Development*, 49(1-2):133–158.

Ritort, F. and Sollich, P. (2003). Glassy dynamics of kinetically constrained models. *Advances in Physics*, 52(4):219–342.

Roelink, H., Porter, J. A., Chiang, C., Tanabe, Y., Chang, D. T., Beachy, P. A., and Jessell, T. M. (1995). Floor plate and motor neuron induction by different concentrations of the amino-terminal cleavage product of sonic hedgehog autoproteolysis. *Cell*, 81(3):445–455.

Rothschild, J. B., Tsimiklis, P., Siggia, E. D., and Francois, P. (2016). Predicting Ancestral Segmentation Phenotypes from Drosophila to Anopheles Using In Silico Evolution. *PLoS Genetics*, 12(5):e1006052.

Rubin, K. J., Lawler, K., Sollich, P., and Ng, T. (2014). Memory effects in biochemical networks as the natural counterpart of extrinsic noise. *Journal of Theoretical Biology*, 357:245–267.

Rubin, K. J. and Sollich, P. (2016). Michaelis-Menten dynamics in protein subnetworks. *Journal of Chemical Physics*, 144(17):174114.

Sagner, A. and Briscoe, J. (2017). Morphogen interpretation: concentration, time, competence, and signaling dynamics. *Wiley Interdisciplinary Reviews: Developmental Biology*, 6(4):e271.

Sagner, A., Gaber, Z. B., Delile, J., Kong, J. H., Rousso, D. L., Pearson, C. A., Weicksel, S. E., Melchionda, M., Mousavy Gharavy, S. N., Briscoe, J., and Novitch, B. G. (2018). Olig2 and Hes regulatory dynamics during motor neuron differentiation revealed by single cell transcriptomics. *PLOS Biology*, 16(2):e2003127.

Sanchez, A., Garcia, H. G., Jones, D., Phillips, R., and Kondev, J. (2011). Effect of promoter architecture on the cell-to-cell variability in gene expression. *PLoS Computational Biology*, 7(3):e1001100.

Sasai, N., Kutejova, E., and Briscoe, J. (2014). Integration of Signals along Orthogonal Axes of the Vertebrate Neural Tube Controls Progenitor Competence and Increases Cell Diversity. *PLoS Biology*, 12(7):1–20.

Schnoerr, D., Sanguinetti, G., and Grima, R. (2017). Approximation and inference methods for stochastic biochemical kinetics - A tutorial review. *Journal of Physics A: Mathematical and Theoretical*, 50(9):093001.

Sharpe, J. (2017). Computer modeling in developmental biology: growing today, essential tomorrow. *Development*, 144(23):4214–4225.

Shea, M. A. and Ackers, G. K. (1985). The ORcontrol system of bacteriophage lambda. A physical-chemical model for gene regulation. *Journal of Molecular Biology*, 181(2):211–230.

Sherman, M. S. and Cohen, B. A. (2012). Thermodynamic state ensemble models of cis-regulation. *PLoS Computational Biology*, 8(3):e1002407.

Smith, S. and Grima, R. (2016). Breakdown of the reaction-diffusion master equation with nonelementary rates. *Physical Review E*, 93(5).

Stinis, P. (2006). A comparative study of two stochastic mode reduction methods. *Physica D: Nonlinear Phenomena*, 213(2):197–213.

Strogatz, S. (2014). *Nonlinear Dynamics and Chaos (Studies in Nonlinearity)*. Advanced book program. Westview Press, 2nd edition.

Stuart, A. M. (1994). Numerical analysis of dynamical systems. *Acta Numerica*, 3:467–572.

Süel, G. M., Garcia-Ojalvo, J., Liberman, L. M., and Elowitz, M. B. (2006). An excitable gene regulatory circuit induces transient cellular differentiation. *Nature*, 440(7083):545–50.

Sunnaker, M., Cedersund, G., and Jirstrand, M. (2011). A method for zooming of nonlinear models of biochemical systems. *BMC Systems Biology*, 5(1):140.

Thomas, P., Grima, R., and Straube, A. V. (2012a). Rigorous elimination of fast stochastic variables from the linear noise approximation using projection operators. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 86(4):041110.

Thomas, P., Straube, A. V., and Grima, R. (2012b). The slow-scale linear noise approximation: An accurate, reduced stochastic description of biochemical networks under timescale separation conditions. *BMC Systems Biology*, 6(ii):39.

Thompson, D. (1917). *On growth and form*.

Tikhomirov, V. M. (1991). *Selected Works of AN Kolmogorov: Volume I: Mathematics and Mechanics*, volume 25. Springer Science & Business Media.

Tokuyama, M. and Mori, H. (1976). Statistical-Mechanical Theory of Random Frequency Modulations and Generalized Brownian Motions. *Progress of Theoretical Physics*, 55(2):411–429.

Tonn, M. K., Thomas, P., Barahona, M., and Oyarzún, D. A. (2019). Stochastic modelling reveals mechanisms of metabolic heterogeneity. *Communications Biology*, 2(1):108.

Turing, A. M. (1952). The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 237(641):37–72.

Umulis, D. M., Shimmi, O., O'Connor, M. B., and Othmer, H. G. (2010). Organism-Scale Modeling of Early Drosophila Patterning via Bone Morphogenetic Proteins. *Developmental Cell*, 18(2):260–274.

Van Kampen, N. G. (2007). *Stochastic Processes in Physics and Chemistry*. North-Holland Personal Library. Elsevier, 3rd ed edition.

Venturi, D. and Karniadakis, G. E. (2014). Convolutionless Nakajima-Zwanzig equations for stochastic analysis in nonlinear dynamical systems. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 470(2166):20130754.

Verd, B., Crombach, A., and Jaeger, J. (2014). Classification of transient behaviours in a time-dependent toggle switch model. *BMC Systems Biology*, 8(1):43.

Verd, B., Monk, N. A., and Jaeger, J. (2019). Modularity, criticality, and evolvability of a developmental gene regulatory network. *eLife*, 8.

von Bertalanffy, L. (1928). *Kritische Theorie der Formbildung,*. Gebrüder Borntraeger, Berlin.

Waddington, C. H. (1940). *Organisers and Genes by C. H. Waddington*. Cambridge biological studies. The University Press.

Wang, J., Zhang, K., Xu, L., and Wang, E. (2011). Quantifying the Waddington landscape and biological paths for development and differentiation. *Proceedings of the National Academy of Sciences of the United States of America*, 108(20):8257–62.

Wang, L., Walker, B. L., Iannaccone, S., Bhatt, D., Kennedy, P. J., and Tse, W. T. (2009). Bistable switches control memory and plasticity in cellular differentiation. *Proceedings of the National Academy of Sciences of the United States of America*, 106(16):6638–43.

Weinan, E., Li, T., and Vanden-Eijnden, E. (2008). Optimal partition and effective dynamics of complex networks. *Proceedings of the National Academy of Sciences of the United States of America*, 105(23):7907–7912.

Wilhelm, T. (2009). The smallest chemical reaction system with bistability. *BMC Systems Biology*, 3(1):90.

Wittkowski, R., Löwen, H., and Brand, H. R. (2012). Extended dynamical density functional theory for colloidal mixtures with temperature gradients. *The Journal of Chemical Physics*, 137(22):224904.

Wolpert, L. (1968). The French flag problem: a contribution to the discussion on pattern development and regulation. *Towards a theoretical biology*, 1:125–133.

Wolpert, L. (1969). Positional information and the spatial pattern of cellular differentiation. *Journal of Theoretical Biology*, 25(1):1–47.

Yu, H., Luscombe, N. M., Qian, J., and Gerstein, M. (2003). Genomic analysis of gene expression relationships in transcriptional regulatory networks. *Trends in Genetics*, 19(8):422–427.

Zhang, L., Radtke, K., Zheng, L., Cai, A. Q., Schilling, T. F., and Nie, Q. (2012). Noise drives sharpening of gene expression boundaries in the zebrafish hindbrain. *Molecular Systems Biology*, 8(613):613.

Zhou, Q. and Anderson, D. J. (2002). The bHLH transcription factors OLIG2 and OLIG1 couple neuronal and glial subtype specification. *Cell*, 109(1):61–73.

Zwanzig, R. (1961). Memory effects in irreversible thermodynamics. *Physical Review*, 124(4):983–992.

Zwanzig, R. (2001). *Nonequilibrium Statistical Mechanics*. Oxford University Press.

A

Appendices

A.1. Appendix: Model parameters

We detail the parameters used throughout the thesis to model neural tube development and adapted for the computational screen as explained in Chapter. 6.

Name	Meaning	Value	Source
α_{P}	Pax6 production rate	2	(Cohen et al., 2014)
α_{O}	Olig2 production rate	2*10	(Cohen et al., 2014) & Sec. 5.2
α_{N}	Nkx2.2 production rate	2	(Cohen et al., 2014)
α_{I}	Irx3 production rate	2	(Cohen et al., 2014)
β_{P}	Pax6 degradation rate	2	(Cohen et al., 2014)
β_{O}	Olig2 degradation rate	2	(Cohen et al., 2014)
β_{N}	Nkx2.2 degradation rate	2	(Cohen et al., 2014)
β_{I}	Irx3 degradation rate	2	(Cohen et al., 2014)
k_{PO}	Olig2 binding to Pax6 DNA	1.9/10	(Cohen et al., 2014) & Sec. 5.2
k_{PN}	Nkx2.2 binding to Pax6 DNA	26.7	(Cohen et al., 2014)
k_{ON}	Nkx2.2 binding to Olig2 DNA	60.6	(Cohen et al., 2014)
k_{OI}	Irx3 binding to Olig2 DNA	28.4	(Cohen et al., 2014)
k_{NP}	Pax6 binding to Nkx2.2 DNA	4.8	(Cohen et al., 2014)
k_{NO}	Olig2 binding to Nkx2.2 DNA	27.1/10	(Cohen et al., 2014) & Sec. 5.2
k_{NI}	Irx3 binding to Nkx2.2 DNA	47.1	(Cohen et al., 2014)
k_{IO}	Olig2 binding to Irx3 DNA	58.8/10	(Cohen et al., 2014) & Sec. 5.2
k_{IN}	Nkx2.2 binding to Irx3 DNA	76.2	(Cohen et al., 2014)
$w_{\mathrm{P,p}}$	Polymerase binding to Pax6 DNA	3.84	(Cohen et al., 2014)
$w_{\mathrm{O,p}}$	Polymerase binding to Olig2 DNA	2.01263	Converted from (Cohen et al., 2014)
$w_{\mathrm{N,p}}$	Polymerase binding to Nkx2.2 DNA	0.572324	Converted from (Cohen et al., 2014)
$w_{\mathrm{I,p}}$	Polymerase binding to Irx3 DNA	18.72	(Cohen et al., 2014)

Name	Meaning	Value	Source
$k_{\mathrm{O,in}}$	Gli (Shh signal) binding to Olig2 DNA	180	Converted from (Cohen et al., 2014)
$k_{\mathrm{N,in}}$	Gli (Shh signal) binding to Nkx2.2 DNA	373	Converted from (Cohen et al., 2014)
Ω	System volume	250	Sec. 5.2
$x_{\mathrm{P}}(0)$	Pax6 initial condition	0.1	Sec. 5.6
$x_{\mathrm{O}}(0)$	Olig2 initial condition	0	Sec. 5.6
$x_{\mathrm{N}}(0)$	Nkx2.2 initial condition	0	Sec. 5.6
$x_{\mathrm{I}}(0)$	Irx3 initial condition	0.1	Sec. 5.6

Where factors of 10 have been written in the table, these arise because we have modified the model of (Cohen et al., 2014) to represent explicitly the experimental observation that Olig2 has a concentration 10 times higher than the other TFs. While this difference is immaterial for a deterministic description of the GRN dynamics, it affects the stochastic representation because larger copy numbers have smaller relative fluctuations.

The above parameters are used in the general model (5.1a) for the dynamics of the TFs $j = $ P (Pax6), O (Olig2), N (Nkx2.2) and I (Irx3). DNA conformations are defined by the numbers $\boldsymbol{n} = (n_{\mathrm{p}}, n_{\mathrm{in}}, n_{\mathrm{P}}, n_{\mathrm{O}}, n_{\mathrm{N}}, n_{\mathrm{I}})$ of bound molecules of polymerase, Gli signal input, Pax6, Olig2, Nkx2.2, Irx3 in that order. The only allowed conformations are the empty conformation, the conformations with polymerase and $n_{\mathrm{in}} = 0$ or 1 signal molecule bound; and conformations with at least one molecule of the other TFs bound, with maximally two molecules from each other TF. All other conformations are assigned affinity zero. The weights for the allowed conformations are multiplicative, with bound polymerase contributing a factor $w_{j,\mathrm{p}}$ (see below), bound signal a factor $k_{j,\mathrm{in}}x_{\mathrm{in}}$ and each TF i bound to DNA producing TF j a factor $k_{ji}x_i$. Examples of the corresponding affinities are $k_{\mathrm{O},(0,0,0,0,1,0)} = k_{\mathrm{ON}}$ and $k_{\mathrm{O},(0,0,0,0,0,2)} = {k_{\mathrm{OI}}}^2$. The

polymerase binding parameters are directly stated as the weights $w_{j,\mathrm{p}} = k_{j,\mathrm{p}} x_{\mathrm{p}}$ including polymerase concentration (which is assumed constant). As detailed in (Cohen et al., 2014), this weight describes all basal production inputs for each TF and thus represents input from TFs such as Sox2. Finally, the protein production rates $\alpha_{j,\boldsymbol{n}}$ in the general model (5.1a) are set to the value given in the table for the DNA conformations with bound polymerase, and zero otherwise.

Explicitly, the production rate for Olig2 is then written as :

$$\frac{\alpha_{\mathrm{O}} w_{\mathrm{O,p}} (1 + k_{\mathrm{O,in}} x_{\mathrm{in}})}{w_{\mathrm{O,p}} (1 + k_{\mathrm{O,in}} x_{\mathrm{in}}) + (1 + k_{\mathrm{OI}} x_{\mathrm{I}})^2 (1 + k_{\mathrm{ON}} x_{\mathrm{N}})^2} \tag{A.1.1}$$

The signal input concentration x_{in} is the gradient $e^{-s/0.15}$, which depends on the dorsal-ventral neural tube position s ranging from 0 to 1 as in (Cohen et al., 2014).

A.2. Appendix: Experimental materials and methods

Experimental results were obtained by Katherine Exelby as detailed in (Exelby et al., 2019) unless specified otherwise.

A.2.1. Mouse Strains

Mouse strains containing the following alleles were used: Pax6(Sey) Ericson et al. (1997) and O2e33 in strain backgrounds C57BL/6Jax and F1(B6xCBA) respectively. The O2e33 allele was derived using zygote injection of CRISPR gRNA and Cas9 plasmids (see below). Embryos were transferred to psuedopregnant females and subsequent pups were genotyped. O2e33 mice were maintained as a heterozygous population; the line was sub-viable with less than 2/40 homozygous offspring surviving. Embryos for analyses were collected at the indicated time points following a mating, with the day of plug detection designated e0.5. All animal procedures were carried out in accordance with the Animal (Scientific Procedures) Act 1986 under the Home Office project licence PPL80/2528 and PD415DD17.

A.2.2. Embryonic Stem Cell Culture

For the enhancer deletion *in vitro*, mouse ES cells containing a fluorescent reporter cotranslated with Olig2 (Olig2::T2A-mKate2) Sagner et al. (2018) were used. Mouse embryonic stem cells were maintained on mitotically inactivated fibroblasts (feeder cells) in ES medium with 1,000 U/ml LIF. Cells were differentiated to spinal cord neural progenitors as previously described Gouti et al. (2014). To initiate differentiation, ES cells were dissociated using 0.05% Trypsin (Gibco) and panned in ES medium on culture plates for 2x 15 minutes to remove feeder cells. ES cells were collected, spun down and re-suspended in N2B27 medium. 50,000 cells were plates on 35mm CellBIND dishes

(Corning). Dishes had been coated with 0.1% gelatine in PBS before addition of 1.5ml of N2B27 with 10 ng/ml bFGF. After 48 hours medium was replaced with N2B27 + 10ng/ml bFGF + 5uM CHIR99021 (Axon). 24 hours later, at D3, medium was replaced with N2B27 + 100nm RA (Sigma) and 500nm SAG (Calbiochem), this was repeated every 24 hours.

A.2.3. CRISPR/Cas9 targeting

For CRISPR/Cas9-mediated excision of the -33 kb enhancer, two pairs of short guide RNA (sgRNA) sequences were designed to target either side of the enhancer region. ZiFit online tool (http://zifit.partners.org/) was used to select guides that had the lowest number of potential off target sites. sgRNA sequences (ACTTTGTAAGCCGAGCC) and (GATAATCGCCTCCCTCC were cloned into pX459 v2.0 (Addgene, Ran et al. (2013)) and transfected into ES cells via nucleofection. This generated a cell line with a 995bp deletion (chr16: 91192464-91193458). Two separate clones were analysed to determine whether there was substantial clonal variation. A second line was generated with a larger deletion of approximately 3.3kb using sgRNA sequences (GTTTATGGCTCATCCCC and TCCAGGCTCCCATATCC). Cell lines with this larger deletion yielded the same results as the smaller deletion (data not shown). To generate the mouse line, plasmids encoding the sgRNAs for the 3.3kb deletion were injected into zygotes before being transferred to pseudo-pregnant females. The mouse line generated had a 3259 bp deletion (chr16: 91191295-91194570).

To assess Olig2 protein copy number, a transgenic cell line was constructed, Olig2-HA-SnapTag. Sequencing encoding an HA tagged SnapTag was placed at the C-terminus of the endogenous coding sequence for Olig2 via homologous recombination using CRISPR. The SnapTag sequence was extracted from the pSNAPf vector (N9183S, NEB) and inserted into a plasmid containing Olig2 Sagner et al. (2018) and targeted as previously

described.

A.2.4. Protein Copy Number Quantification

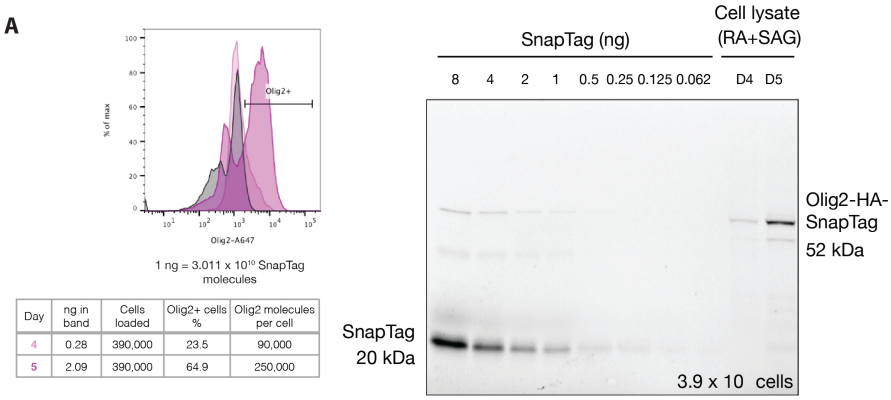
All work relating to protein copy number quantification was performed by Lorena Garcia Perez as stated in (Exelby et al., 2019).

The concentration of recombinant proteins (used as standards) was calculated from Coomassie staining (GelCode Blue Stain Reagent, Thermo scientific). Recombinant proteins used were Pax6 (Bioclone, PI-0099) Nkx2.2 (MyBioSource, MBS717917) and SnapTag (NEB, P9312S). A solution of 5 m SNAP-tag was labelled with Janelia Fluor JF549 (TOCRIS, 6147) SnapTag Ligand at 10 m (assembled in house) for 30 mins at 37°C.

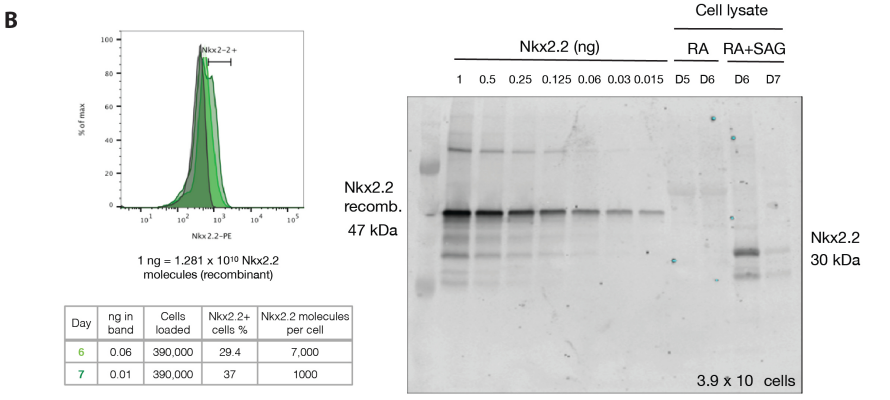
To determine Pax6 and Nkx2.2 average molecule number per cell, a WT HM1 mouse embryonic stem cell line was used Doetschman et al. (1987). Cells were lysed in RIPA buffer supplemented with protease inhibitors. The cell lysates were analysed by Western blot, with lysate from a known number of cells loaded per lane. The following antibodies were used: rabbit anti-Pax6 (Millipore AB2237, 1:2000), mouse anti-Nkx2.2 (DSHB 745A5, 1:50), donkey anti-mouse IRDye 800CW (Licor) and donkey anti-rabbit IRDye 680RD (Licor). Blots were scanned using an Odyssey Scanner (Licor).

We used the cell line Olig2-HA-SnapTag to determine protein copy number for Olig2. Cells for Olig2 and Nkx2.2 copy numbers were differentiated as described. For Pax6, cells were exposed to 100nm RA only from day 4 to induce a more dorsal spinal cord cell fate. One day prior to sample collection, the cells were incubated with Janelia Fluor JF549 SnapTag Ligand (assembled in house) directly in the media at 1 μM overnight. Cells were lysed in RIPA buffer supplemented with protease inhibitors. A known number of cells were loaded per lane. Gels were scanned using Typhoon FLA 9500. To determine the percentage of expressing cells, flow cytometry was carried out as described in the

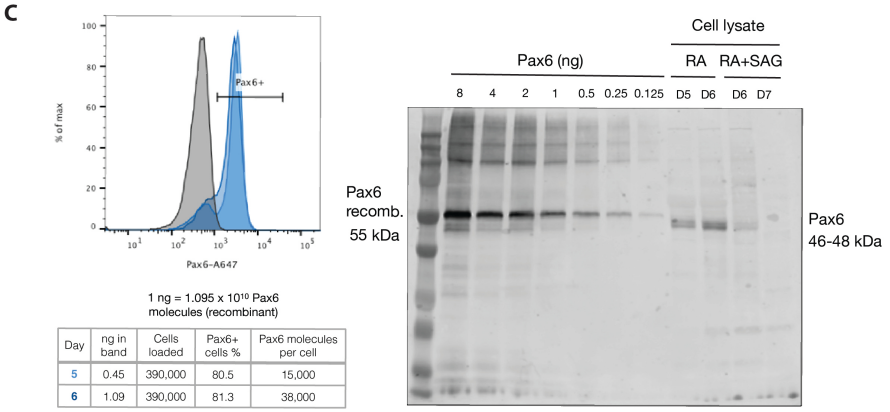
Flow Cytometry section.



Day	ng in band	Cells loaded	Olig2+ cells %	Olig2 molecules per cell
4	0.28	390,000	23.5	90,000
5	2.09	390,000	64.9	250,000



Day	ng in band	Cells loaded	Nkx2.2+ cells %	Nkx2.2 molecules per cell
6	0.06	390,000	29.4	7,000
7	0.01	390,000	37	1000



Day	ng in band	Cells loaded	Pax6+ cells %	Pax6 molecules per cell
5	0.45	390,000	80.5	15,000
6	1.09	390,000	81.3	38,000

Figure A1: Quantifying Protein Copy Number (**A**) Flow cytometry analysis to determine percentage of Olig2 expressing cells in differentiated ES cells at the indicated days. Table shows quantification of a gel for days 4 and 5. Olig2 has approximately a 10-fold higher protein copy number compared to Nkx2.2 and Pax6. (**B**) Analysis of Nkx2.2 expressing cells on days 6 and 7 of differentiation. Nkx2.2 molecules per cell calculated using the measured percentage of cells expressing Nkx2.2 and quantification of the Western blot analysis. (**C**) Analysis of Pax6 expressing cells to determine protein copy number at days 5 and 6 of differentiation. Pax6 molecules per cell calculated using the measured percentage of cells expressing Pax6 and quantification of the Western blot analysis.

A.2.5. Flow Cytometry Analysis

Cells were dissociated using 0.05% Trypsin and collected in ES media. Cells were then washed in PBS and resuspensed in PBS containing live-cell Calcein Violet dye (Life Technologies). Control and O2e33 cells were differentiated in parallel and analysed together. Control cells differentiated without SAG from day 4 were used to set population gates for mKate positive cells.

For protein quantifications, flow cytometry was used to determine percentage of cells expressing Olig2, Pax6 and Nkx2.2. Cells were labelled with either PE Mouse anti-Nkx2.2 (BD Pharmingen 564730, 1:20); AlexaFluor 647 mouse anti-Human Pax6 (BD Pharmingen 562249, 1:50); goat anti-Olig2 (R&D Systems AF2418, 1:800) then donkey anti-goat 405 (Biotium 20398, 1:500). Flow analysis was performed using a Becton Dickinson LSRII flow cytometer.

A.2.6. Immunohistochemistry and Microscopy

Embryos were collected at defined timepoints and fixed for 30 minutes for e8.5, 1 hour for e9.5 and 2 hours for e10.5 in 4% paraformaldehyde in PBS. Embryos for wholemount imaging were washed in PBS containing 0.1% Triton X-100 (PBST) before addition of primary antibodies. Embryos for sectioning were placed in cryopreservation 30% sucrose overnight at 4°C then dissected into forelimb neural tube fragments. These

were mounted in gelatine then frozen. 12 m sections were collected on glass slides using Zeiss Hyrax C 60R cryostat. Gelatine was removed from the slides by 4 x 5 min washes in PBS at 42°C and sections washed with PBST. For *in vitro* stainings, cells were washed in PBS and fixed in 4% paraformaldehyde for 15 min at 4°C then washed in PBS then PBST. For whole embryos, embryo sections and cells, primary antibodies diluted in blocking solution (1% BSA in PBST) were applied overnight at 4°C. These were then washed in 3 x PBST before secondary antibodies diluted in PBST were added for 1 hour at room temperature. Secondary antibodies were removed with 3 x washes with PBST and one wash containing PBST and DAPI. Sections and cells were mounted using Prolong Gold (Invitrogen). Embryos for wholemount were mounted using glycerol. Primary antibodies used were guinea pig anti-Olig2 (gift from Bennett Novitch, 1:8000 Novitch et al. (2001)); mouse anti-Nkx2.2 (BD Pharmingen 564731, 1:500); rabbit anti-Pax6 (Millipore AB2237, 1:1000); goat anti-Sox2 (R&D Systems AF2018, 1:200). All secondary antibodies were raised in donkey and conjugated to Alexa488, Alexa568, Alexa647 (Abcam).

Cells were imaged on a Zeiss Imager.Z2 microscope using 20x objective. Z-stacks were taken and presented as a maximum projection using FiJi imaging software. A Leica SP5 upright confocal microscope was used to image embryo sections (40x oil objective) and whole embryos (20x dry objective). For whole embryos, z-stacks were taken across a tile-scan then assembled and maximally projected using FiJi imaging software.

A.2.7. Supplementary Figures

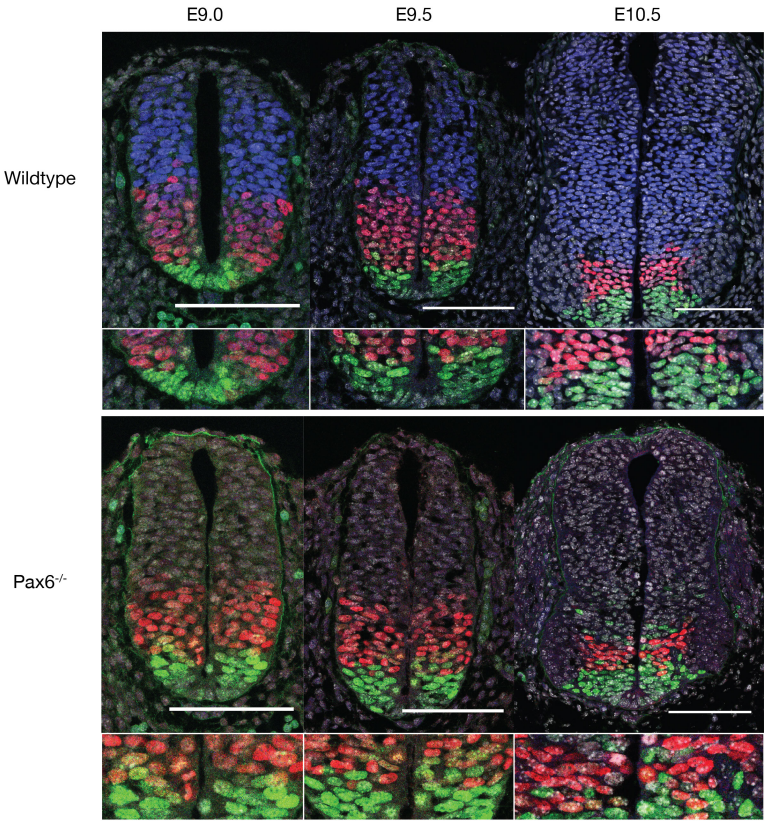


Figure A2: pMN-p3 boundary precision decreases over time in Pax6 mutants. Transverse sections of wildtype and Pax6$^{-/-}$ embryos between e9.0 and e10.5 stained for Pax6 (blue), Olig2 (red) and Nkx2.2 (green). Scale bar = 100μm. The pMN-p3 boundary becomes less well defined at later time points.

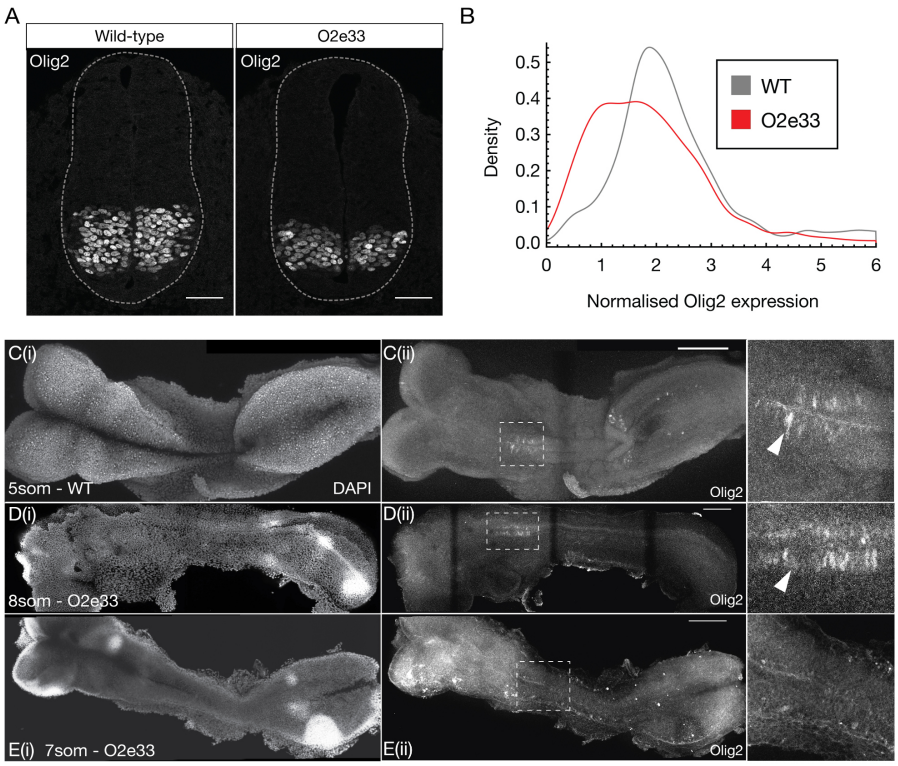


Figure A3: Olig2 expression in O2e33 mutants is lower and delayed in onset. (**A**) Transverse brachial sections of e9.5 WT and O2e33 embryos stained for Olig2. The O2e33 embryo has a smaller Olig2 domain with reduced expression levels. Scale bar = 50μm (**B**) Normalised Olig2 expression for single cells in WT and O2e33 embryo sections. (**C, D, E**) Wholemount images of WT (C) and O2e33 mutants (D, E) for DAPI (i) and Olig2 staining (ii-iii). Expression of Olig2 in wildtype is observed at 5 somites but in O2e33 Olig2 onset occurs later at 8 somites. Olig2 is not observed in O2e33 embryos at 7 somites. Scale bar = 100μm

A.3. Appendix: Image quantification

The analysis performed in this appendix was performed entirely by me, using the data acquired by Katherine Exelby for (Exelby et al., 2019)

A.3.1. Fluorescent intensity measurements

Single optical planes from confocal z-stack images were used for analysis. Each nucleus was identified individually using the FiJi point tool. The DAPI channel was used as reference for the position of the nuclei regardless of TF expression. A circle of 2 μm radius was taken around each point, x and y position and mean fluorescence intensity values for Nkx2.2, Olig2 and Pax6 were recorded. Reference points at the ventral and dorsal pole of the neural tube in each section were recorded in order to align all embryos along the dorso-ventral axis.

A.3.2. Pre-processing

We performed a set of normalisation steps in order to compare embryos from different batches and across phenotypes:

1. The datasets were realigned vertically with respect to the reference points and the ventral-most point was set to (0,0) in axes coordinates

2. Cells with DAPI levels below two SDs from the mean were removed to eliminate falsely identified nuclei. This value was decided individually for each sample to account for different background levels resulting from technical noise.

3. Points that were very low in intensity (below two SDs) were set to a minimum threshold in each individual channel.

4. For Nkx2.2 and Olig2, the intensity values were re-scaled such that the minimum value is at 0 and the 40% quantile is at the arbitrary value of 0.08. This was done individually for each embryo with the assumption that most nuclei in a full neural tube cross-section will not express these proteins.

5. For Pax6, most nuclei in the image express some level of Pax6; accordingly we set the 60% quantile at 0.6 across all datasets.

A.3.3. Staging embryos with size

We used the dorsal-ventral length of the neural tube as a proxy for embryo age Cohen et al. (2015). For e9.5 embryos, the neural tube size measured was between 250μm and 350μm and for e10.5 embryos were larger than 350μm. In order to subgroup e9.5 embryos, neural tube size was used. In total we have 46 WT, 29 O2e33 and 16 Pax6$^{-/-}$. By sizes they are distributed as:

	WT	O2e33	Pax6$^{-/-}$
150 - 250 μm	17	5	5
250 - 350 μm	13	13	3
350 - μm	16	11	8

A.3.4. Classification into cell types

In order to analyse the heterogeneity at the boundary between domains, we classified all cells into one of 5 specific cell types: floor plate, p3, pMN, Irx3 positive, other; this was done based on the position and expression profile of each cell. We refrained from using the Pax6 channel in our classifier to avoid any bias in the classification of Pax6$^{-/-}$ embryos. We therefore classified based on three parameters: Nkx2.2 intensity, Olig2 intensity and dorsal-ventral position. The thresholds we employed for Nkx2.2 and Olig2

concentrations are shown in Fig. A4**A**-**B**. There was a further constraint on the dorsal-ventral position for each cell type, in order to avoid anomalies from blood vessels and imaging artefacts and to be able to separate floor plate cells from Irx3 positive cells, both of which lack expression of Nkx2.2 and Olig2 (Fig. A4**B**-**C**). Manually bench-marking this method indicated that we were able to classify most cells accurately for all three phenotypes. The classifier becomes less accurate for cells in dorsal regions but this is of no concern as our subsequent analysis did not involve these cells. For the specific task of quantifying the Olig2-Irx3 boundary position we employed the Pax6 channel as a further parameter to aid classification. This was only performed for WT and O2e33 (data not shown).

A.3.5. Defining boundary position and width

Once the cell types had been classified we assigned a quantitative measure of the width of gene expression boundaries. For this we fit to the cell position data, for each embryo, a smooth function indicating the probability of finding a cell of one type (the prevalent type on one side of the boundary) at each location of the image. We focused on the boundary between p3 and pMN domains. The classifier is then binary and gives the probability of finding a p3 cell at each image location. We used a Gaussian process approach to fit this classifier as detailed in Rasmussen and Williams (2004), using public MATLAB code (MATLAB version r2018b). The Gaussian process was chosen to have a constant mean function and a squared exponential covariance function. This choice of covariance function is relatively standard and allows us in particular to assign separate covariance function lengthscales in the x and y image directions by automatic relevance determination Rasmussen and Williams (2004). We used a logistic transfer function to convert Gaussian process values to probabilities, again a standard choice. Once the classification probabilities have been obtained in this way, we define the boundary as the

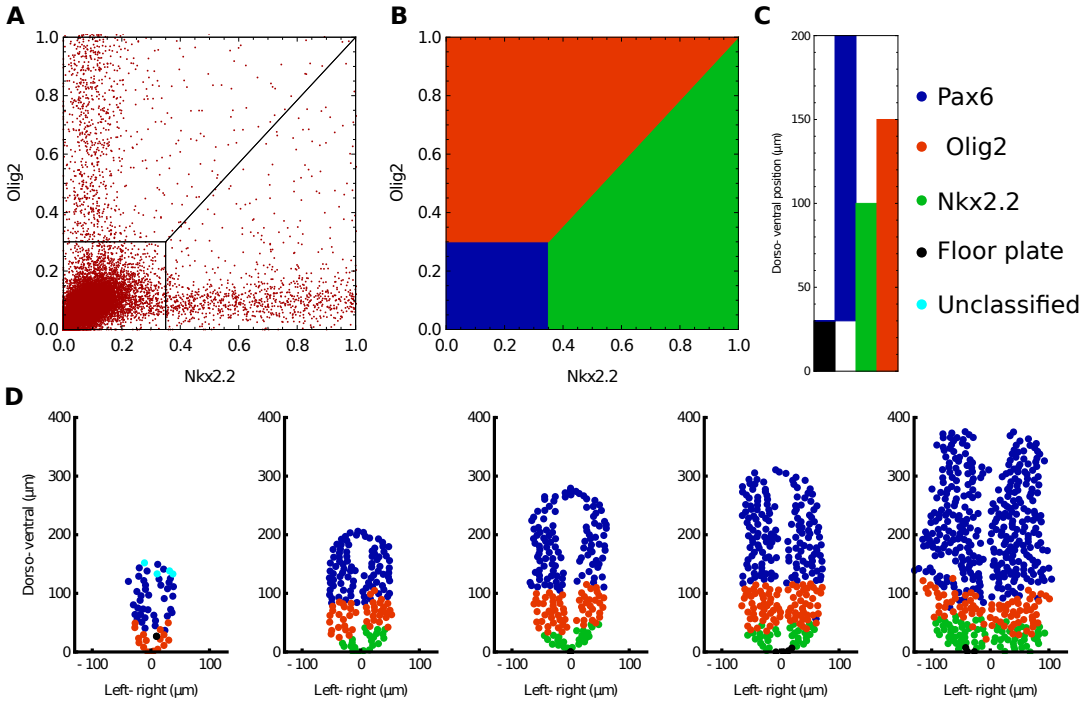


Figure A4: Analysis of gene expression in embryos(**A**) Plot illustrating the concentrations of Nkx2.2 and Olig2 for all cells analysed. This highlights that the majority of cells are negative for both TFs and also that very few cells co-express both TFs. (**B**) Criteria to determine the identity of each cell by using the levels of Nkx2.2 and Olig2; colours indicating cell assignment as Olig2 (red), Nkx2.2 (green) and neither (blue) are consistent throughout the figure. The concentration of Pax6 is not used for classification. (**C**) Positional limits along the neural tube for each cell type. Cells that express neither Olig2 nor Nkx2.2 are classified based on their position as they can be ventral floor plate cells (black) or more dorsal progenitors. Cells that have mismatching values of concentration and position are classified as exceptions in Cyan (**D**) Examples of classified embryos of increasing age, illustrating the accuracy of the approach for determining cell type.

region where the probability of p3 cells lies in the range 11% to 89%, i.e. where there is significant mixing of cell types. We then determine the width of this region geometrically. This method allowed us to calculate the boundary widths for all embryos in a consistent manner, and to compare WT with mutants. The boundary region is determined from the trained classifier for each embryo as explained above; the position where the classification probability is 50% for either cell type is used to define the position of the boundary (an average position of the boundary along the left-right axis) (Fig. A5).

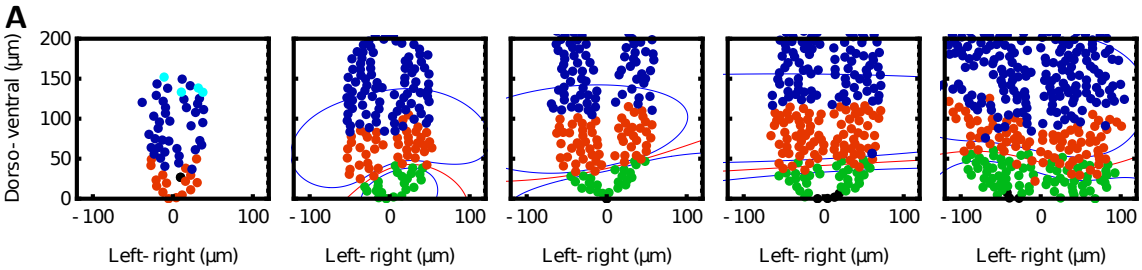


Figure A5: Examples of boundaries determined by the Gaussian process classifier. The red lines indicate the computed boundary position, and correspond to the image locations where the probability of being a p3 or pMN cell is 0.5. Blue lines close to the p3-pMN boundary delimit the area identified as the boundary region, where the probability of being a p3 cell is in the range 11% to 89%. By measuring the area between the two blue curves and dividing by the width of the embryo we are able to quantify the width of the boundaries. In turn by obtaining the average position of the red line, we are able to calculate the boundary position.

A.3.6. Quantifying TF levels

We extracted Olig2 positive cells that were classified as being within the boundary region. The model predicted that these cells were the most likely to transition to a Nkx2.2 positive state, given sufficient time. We quantify the levels of Pax6 and Olig2 for these cells in WT and O2E33 mutants. The resulting measurements do not provide absolute numbers; but given that all samples are normalised in the same way, as described (Sec. A.3.2), the resulting measurements are comparable relative to each other. We use these measurements as equivalents to observing fluctuations around a steady state over a series of dorso-ventral positions. In this way, we take the corresponding equivalent in the simulations, where we also average fluctuations across several neural tube positions (Sec. 5.2).

A.3.7. Calculating variance levels

In order to calculate the total variance of Olig2 and Pax6 levels within the pMN domain we extracted all Olig2 expressing cells, for both WT and O2e33, outside the boundary region. The variances and covariances of the normalised fluorescence intensity values

were calculated, analogous to the theoretical approach (Sec. 5.2). The square root of the trace of the resulting covariance matrices were then used to obtain the typical root-mean-square relative variance.

A.4. Appendix: Calculating magnitude of fluctuations

To compare the magnitude of fluctuations between WT and mutants *in silico* we take two separate approaches. The first is to consider fluctuations in expression levels around a steady state, before any transition to a new state occurs. For moderate noise levels such fluctuations can be analysed using a linear expansion of the dynamics around the steady state (here: pMN), leading to a local Gaussian distribution of expression levels. The corresponding covariance matrix $\boldsymbol{C}$ can be calculated from the Jacobian matrix $\boldsymbol{J}$ of the linearized dynamics and the noise covariance $\boldsymbol{D}$ as defined in (5.1b), both evaluated at the steady state. The required link between the three matrices is the Lyapunov equation, which determines $\boldsymbol{C}$ via

$$\boldsymbol{D} = \boldsymbol{J}\boldsymbol{C} + \boldsymbol{C}\boldsymbol{J}^{\mathrm{T}}$$

Once $\boldsymbol{C}$ has been found we normalise it by the corresponding pMN steady state values ($\boldsymbol{X}$), to obtain $\bar{\boldsymbol{C}} = \mathrm{diag}(\boldsymbol{X})^{-1}\boldsymbol{C}\mathrm{diag}(\boldsymbol{X})^{-1}$. We finally take the trace of $\bar{\boldsymbol{C}}$ and take the square root. The end result is the typical standard deviation (root-mean-square fluctuation) of the expression levels, relative to the mean expression levels, which is shown in Fig. A6**A** as a function of neural tube position.

The second approach to quantifying noise levels is to use the noise variance, which is the trace of the noise covariance matrix given in (5.1b). This noise variance depends on the expression levels so we measure it at equidistant points along the MAP and take the square root of this value to obtain the root-mean-square noise level. Example results at a specific position along the neural tube are shown in Fig. A6**B**; results at other positions were qualitatively the same (data not shown). Both approaches to quantifying noise show comparable total variance across the different genotypes, with slightly lower noise

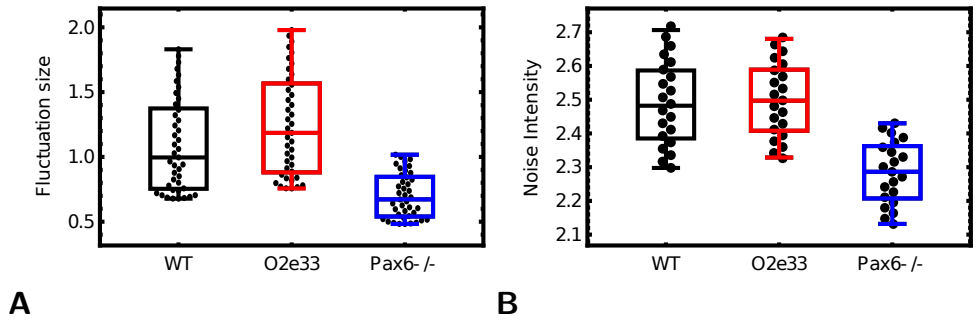


Figure A6: Comparing total noise across genotypes (A) Comparison of noise levels as defined by root-mean-square relative expression level fluctuations, calculated within a Gaussian approximation near the steady state. Points represent different positions along the neural tube (B) Noise levels defined as noise variance calculated at equidistant points along the MAP, at fixed fractional neural tube length from the bifurcation point. Note that in both definitions, noise levels are comparable across WT and both mutants, with slightly lower values in Pax6$-/-$.

in Pax6$^{-/-}$ than in WT and O2e33. To make the comparison to *in vivo* observations we accounted for the fact that experimentally, noise levels are averaged across several neural tube positions throughout the pMN domain. We therefore also performed an average *in silico* of neural tube positions to obtain comparable data for Fig. 5.6**K**.